Vers la conception de documents composites : extraction et organisation de l'information pertinente

Sylvain Lamprier

université angers

Année : 2008
N d'ordre : 952

Vers la conception de documents composites : extraction et organisation de l'information pertinente

Thèse de doctorat

Spécialité : Informatique

École Doctorale STIM, ED 503

Présentée et soutenue publiquement

Le 5 Décembre 2008

À Angers

Par

Sylvain LAMPRIER

Devant le jury ci-dessous :

Président :	Adeline Nazarenko,	Professeur à l'Université Paris-Nord
Rapporteurs :	Mohand Boughanem,	Professeur à l'Université de Toulouse
	Marc El-Bèze,	Professeur à l'Université d'Avignon
Examinateurs :	Xavier Gandibleux,	Professeur à l'Université de Nantes
	Pierre-François Marteau,	Professeur à l'Université de Bretagne Sud
Directeur de thèse :	Frédéric Saubion,	Professeur à l'Université d'Angers
Co-directeur de thèse :	Tassadit Amghar,	Maître de Conférences à l'Université d'Angers
Co-directeur de thèse :	Bernard Levrat,	Professeur à l'Université d'Angers

Laboratoire d'Étude et de Recherche en Informatique d'Angers
2, Bd Lavoisier, 49045 Angers Cedex 01

Remerciements

Au terme du travail de thèse présenté ici, je voudrais exprimer ma reconnaissance envers tous ceux qui, de quelle que manière que ce soit, m'ont aidé à le mener à bien.

En premier lieu, je tiens à remercier très chaleureusement Frédéric Saubion, Tassadit Amghar et Bernard Levrat, mes encadrants, avec qui j' ai pris un très grand plaisir à travailler tout au long de ces trois années. Pour votre disponibilité, votre sympathie, votre implication, vos très nombreux conseils et encouragements... À vous trois, merci beaucoup.

En outre, j'adresse mes plus sincères remerciements à l'ensemble des membres de mon jury :

- à Adeline Nazarenko, pour avoir aimablement accepté de le présider ;
- à Mohand Boughanem et Marc El-Bèze, pour avoir émis, en tant que rapporteurs, leurs avis éclairés sur mes travaux de recherche ;
- et à Xavier Gandibleux et Pierre-François Marteau, pour y avoir activement participé.

Je suis bien conscient que l'appréciation d'un travail tel que celui présenté ici constitue un effort considérable dont je leur suis extrêmement reconnaissant de s'être si consciencieusement acquitté, malgré un calendrier de fin d'année pour beaucoup très chargé.

Je souhaite remercier l'ensemble des membres du LERIA, le laboratoire d'informatique de l'université d'Angers, pour m'avoir si agréablement accueilli, et tout particulièrement Frédéric Lardeux dont les conseils réguliers me furent des plus utiles.

Mes remerciements vont également à mes collègues et amis thésards, Olivier Cantin, Vincent Derrien, Adrien Goëffon, Giglia Gomez, Tony Lambert, Jorge Maturana, Daniel Porumbel et Thomas Raimbault pour les agréables moments passés ensemble. Je tiens à remercier particulièrement Adrien Goëffon, pour m'avoir en outre aidé à résoudre un problème d'optimisation qui me posait quelques difficultés.

Je n'oublie pas mes amis extérieurs à l'université, à qui j'adresse tous mes remerciements pour leur soutien, leur amitié et les excellents moments passés en leur compagnie. La participation de mes très bons amis Evelyne Audouard, Thomas Colin, Cedric Dauphin, Marine Derrien, Charlotte Dugardin, Jean-Francois El-Hajjar, Guilhem Fabre, Fabienne Fily, Vincent Fremaux, Aurélien Gendron, Julien Naroun, Alexandre Pelluault et Rodolphe Quemard à certaines expérimentations reportées dans ce mémoire m'a par ailleurs été d'une aide très précieuse.

Mes remerciements ne seraient pas complets si je n'insistais pas sur le grand mérite revenant à ma très chère Fabienne, tant elle a partagé les doutes et les satisfactions que ce doctorat a pu me procurer. Merci d'avoir su m'encourager, me soutenir et me supporter tout au long de ces trois années.

Enfin, je remercie très affectueusement ma famille, et tout particulièrement mes parents, pour l'amour et la confiance dont ils m'ont toujours témoigné. Pour les nombreux efforts consentis afin que je mène à bien mes études, merci du fond du cœur.

Table des matières

Introduction Générale **1**

Partie I État de l'art sur la recherche d'information

1 Les systèmes de recherche d'information **11**
1.1 Présentation et objectifs . . . 12
1.2 Représentation des textes . . . 13
1.2.1 Lemmatisation . . . 14
1.2.2 Sélection des termes représentatifs . . . 15
1.2.3 Structure d'indexation . . . 18
1.3 Estimation de la pertinence des documents . . . 19
1.3.1 Modèles de recherche . . . 19
1.3.2 Mesures de similarité . . . 31
1.4 Formulation d'une recherche d'information . . . 36
1.4.1 Types de requêtes . . . 37
1.4.2 Expansion de requêtes . . . 38
1.5 Présentation des résultats . . . 41
1.6 Conclusion . . . 45

2 Clustering et recherche d'information **47**
2.1 Présentation et applications . . . 48
2.2 Relations entre documents . . . 52
2.3 Méthodes de clustering . . . 53
2.3.1 Mode de représentation des clusters . . . 55
2.3.2 Méthodes non-hériachiques . . . 57
2.3.3 Méthodes hiérarchiques . . . 60
2.4 Description du contenu des groupes thématiques . . . 63
2.5 Conclusion . . . 66

3 Évaluation des systèmes de recherche d'information **67**
3.1 Enjeux et problématiques . . . 68
3.2 Corpus, requêtes et pertinence . . . 69
3.3 Méthodologies d'évaluation . . . 73
3.3.1 Liste ordonnée de résultats . . . 73
3.3.2 Groupes de documents . . . 76
3.4 Conclusion . . . 81

4 Meta-heuristiques et recherche d'information **83**
4.1 Optimisation combinatoire . . . 84
4.2 Algorithmes génétiques . . . 86
4.3 Optimisation multi-objectifs . . . 90
4.3.1 La dominance au sens de *Pareto* . . . 90
4.3.2 Le *Strength Pareto Evolutionary Algorithm* . . . 93
4.4 Application à la recherche d'information . . . 96
4.5 Conclusion . . . 98

Partie II Extraction des thématiques

5 Segmentation thématique **103**
5.1 La Segmentation thématique de textes . . . 104
5.2 Méthodes de segmentation thématique . . . 105
5.3 Méthodologies d'évaluation . . . 108
5.3.1 Constitution d'une segmentation de référence . . . 108
5.3.2 Mesures d'évaluation . . . 110
5.4 Vers une segmentation globale et cohérente des textes . . . 112
5.4.1 ClassStruggle : mise en concurrence de groupes thématiques . . . 115
5.4.2 SegGen : optimisation multi-objectifs des segments . . . 122
5.4.3 Évaluation des systèmes . . . 127
5.5 Vers un mode d'évaluation plus équitable . . . 133
5.5.1 Analyse de la mesure d'évaluation WindowDiff . . . 134
5.5.2 Prise en compte des risques encourus . . . 139
5.5.3 Respect des différences . . . 142
5.6 Conclusion . . . 147

6 Segmenter pour ordonner les documents **149**
6.1 Les approches de type *Passage Retrieval* . . . 150
6.2 Mesures de pertinence et longueur des textes . . . 152
6.2.1 Impacts de la longueur des textes sur les estimations de pertinence . . . 157
6.2.2 Normalisation des scores de pertinence par régression statistique . . . 160
6.3 Types de segments et performances des systèmes . . . 164

6.3.1 Types de segments étudiés . . . 165
6.3.2 Comparaison des approches . . . 166
6.4 Conclusion . . . 169

7 Segmenter pour regrouper les documents pertinents 171
7.1 Mesures de proximité thématique et longueur des textes . . . 172
7.2 Vers des groupes plus représentatifs des sujets abordés . . . 174
7.2.1 Approches proposées . . . 174
7.2.2 Comparaison des approches . . . 176
7.3 Conclusion . . . 181

Partie III Organisation de l'information pertinente

8 Concentration vs. Distribution de l'information pertinente 185
8.1 Remise en cause de la Cluster Hypothesis . . . 186
8.2 Évaluer l'accès à l'information . . . 189
8.2.1 Parcours optimal . . . 192
8.2.2 Parcours moyen . . . 195
8.2.3 Parcours orienté par la pertinence des documents . . . 197
8.2.4 Parcours orienté par la proximité des documents pertinents . . . 198
8.2.5 Étude des mesures proposées . . . 201
8.3 Comparaison des systèmes . . . 207
8.4 Conclusion . . . 210

9 Présenter un aperçu de l'information pertinente 213
9.1 Un clustering multi-objectifs orienté requête . . . 214
9.2 Évaluation du système . . . 217
9.2.1 Influence du critère de proximité des groupes avec la requête . . . 218
9.2.2 Influence du nombre de documents considérés . . . 221
9.2.3 Influence du nombre de groupes produits . . . 222
9.2.4 Détermination du nombre de groupes optimal . . . 223
9.3 Application aux segments thématiques . . . 224
9.3.1 Influence de la segmentation sur l'organisation des clusters . . . 224
9.3.2 Que présenter à l'utilisateur ? . . . 226
9.4 Conclusion . . . 227

Conclusion Générale 229

Liste des figures 235

Liste des tables **237**

Liste des algorithmes **239**

Index **241**

Références bibliographiques **243**

Résumé / Abstract **278**

Introduction Générale

Au vu du nombre sans cesse croissant de documents électroniques disponibles sur Internet et dans les bases de données, retrouver des informations correspondant à un besoin est bien souvent considéré comme un processus cognitif très complexe, qui fait appel à de nombreux savoirs et se compose de diverses tâches, allant de la prise en compte du manque d'information jusqu'au traitement des données identifiées [Marchionini, 1995]. La recherche d'information (RI), branche de l'informatique s'intéressant à l'acquisition, l'organisation, le stockage, la recherche et la sélection d'informations [Salton, 1968], a pour objectif principal de concevoir des systèmes permettant d'aider des utilisateurs à trouver les informations qui les intéressent malgré la masse de données disponibles. Au sens large, la recherche d'information inclut deux aspects : l'indexation des corpus et l'interrogation du fonds documentaire ainsi constitué. À première vue, l'élaboration de systèmes de recherche semble alors simple : l'utilisateur pose une question au système qui balaye l'index pour en retourner les documents qui y répondent le mieux. Néanmoins, de nombreux problèmes se posent, notamment en ce qui concerne la formulation de la recherche, la représentation des textes ou la présentation des résultats. En effet, comment exprimer un besoin d'information qui puisse être compréhensible par le système ? Comment formuler de manière précise sa recherche alors qu'il nous manque justement des informations sur le sujet ? Comment caractériser le contenu des documents afin de pouvoir le rapprocher de la question posée ? Comment estimer la pertinence des documents ? Sous quelle forme produire les résultats de la recherche pour qu'ils soient le plus facilement interprétables par l'utilisateur ? Autant de questions, parmi de nombreuses autres, qui font de l'élaboration des systèmes de recherche d'information le siège d'intenses réflexions depuis des décennies.

Dans cette thèse, nous nous intéressons principalement au problème de présentation des résultats d'un système de recherche d'information. L'objectif est de permettre à un utilisateur de cerner rapidement les différents aspects de la requête formulée. Typiquement, un système de recherche d'information retourne, en réponse à la requête d'un utilisateur, une liste de documents ordonnée selon une estimation de leur potentiel de pertinence [Voorhees and Harman, 1997]. Cette organisation des résultats, qui est employée par la majorité des systèmes de recherche, impose à l'utilisateur de parcourir linéairement la liste de résultats, en examinant les documents un à un jusqu'à avoir le sentiment d'avoir collecté suffisamment d'informations. Outre le fait qu'un tel parcours risque de s'avérer très fastidieux, tout le problème est de savoir quand s'arrêter. À partir de quel moment est-on certain d'avoir récolté assez d'informations ? L'utilisateur doit prendre la décision d'arrêter la collecte d'informations alors qu'il ne connait pas la diversité des textes en relation avec sa requête. Dans le but de réduire l'effort à fournir pour localiser les information pertinentes, de nombreuses approches ont proposé des présentations alternatives des résultats. Nombre de ces approches s'appuient sur un clustering des documents retournés par un système de recherche initial pour regrouper les documents aux thématiques similaires et ainsi présenter des catégories de résultats permettant une localisation des documents pertinents facilitée

[Hearst and Pedersen, 1996; Tombros *et al.*, 2002]. La mise en évidence des relations entre documents du corpus permet alors de guider l'utilisateur dans sa recherche [Croft, 1980]. De tels systèmes présentent généralement une liste de descriptions des différents groupes produits [Leuski, 2001b], ce qui permet à l'utilisateur de n'avoir qu'à identifier les descriptions qui semblent le mieux correspondre à ses besoins d'information pour s'orienter vers les groupes contenant les informations susceptibles de l'intéresser. Bien que ce mode de présentation des résultats ait, à de maintes reprises, montré sa capacité à améliorer l'accès à l'information, les systèmes de recherche réalisant une catégorisation de leurs résultats souffrent, selon nous, de deux principales limitations :

- Le degré de diversité thématique intra-document influe sur la capacité à produire des groupes représentatifs de thématiques spécifiques : chaque document est susceptible d'aborder un certain nombre de thématiques distinctes et la prise en compte de relations entre documents aux thématiques diverses risque de conduire à l'obtention de clusters mal centrés autour des principaux sujets abordés ; l'ajout d'un document hétérogène à un groupe risquant de faire dévier ce dernier de sa thématique initiale. Par ailleurs, même si un document est fortement connecté à la requête de l'utilisateur, il est possible que les informations pouvant satisfaire les besoins de la recherche soient contenues dans une partie restreinte de son texte. La prise en compte de documents entiers n'est alors peut-être pas le meilleur moyen d'aider un utilisateur dans sa recherche d'information.
- Le degré de diversité thématique inter-documents influe sur la capacité à produire des groupes représentatifs des différents aspects de la requête : le niveau d'hétérogénéité des textes considérés implique bien souvent un faible degré de finesse du clustering réalisé et certaines thématiques émergentes peuvent se trouver en forte déconnexion avec le besoin exprimé par l'utilisateur. La plupart des systèmes réalisant une catégorisation des résultats d'une recherche préliminaire y voient un effet bénéfique puisque cela permet de regrouper la plupart des textes pertinents dans un même cluster, donnant ainsi la possibilité à un utilisateur de filtrer les résultats retournés en ne parcourant que le cluster contenant les informations qui l'intéressent. Néanmoins, un tel regroupement de l'information pertinente ne permet pas de présenter à un utilisateur les différents aspects de sa recherche. De plus, le regroupement des documents pertinents n'est pas toujours évident et certains éléments informatifs sont susceptibles d'être ignorés, ce qui risque alors de restreindre la perception de l'utilisateur à un unique point de vue.

Nous proposons d'explorer ces deux différents points afin de mettre en place un système permettant de présenter à l'utilisateur une liste de représentants de clusters constituant un bon aperçu des différents types d'information qu'il pourra trouver en rapport avec sa requête dans le corpus de textes interrogé. Dans un premier temps, nous cherchons à individualiser les différentes thématiques des documents pour produire des groupes mieux centrés autour de sujets spécifiques. Dans un second temps, nous cherchons à organiser les groupes autour de la requête de l'utilisateur, afin de proposer une catégorisation des résultats permettant d'appréhender facilement la structure de l'information pertinente. L'objectif final est d'extraire les parties les plus intéressantes d'un ensemble de documents (les documents retournés par un système de recherche classique) afin de présenter à l'utili-

sateur une liste de passages de texte lui permettant de sélectionner les aspects, et donc les groupes de passages, qui lui semblent correspondre au mieux à ses besoins. Cette thèse se concentre donc sur le partitionnement des informations qui ont trait à la requête formulée par l'utilisateur pour en faire émerger la structure dans un document composite final qui peut être considéré comme une "feuille de route".

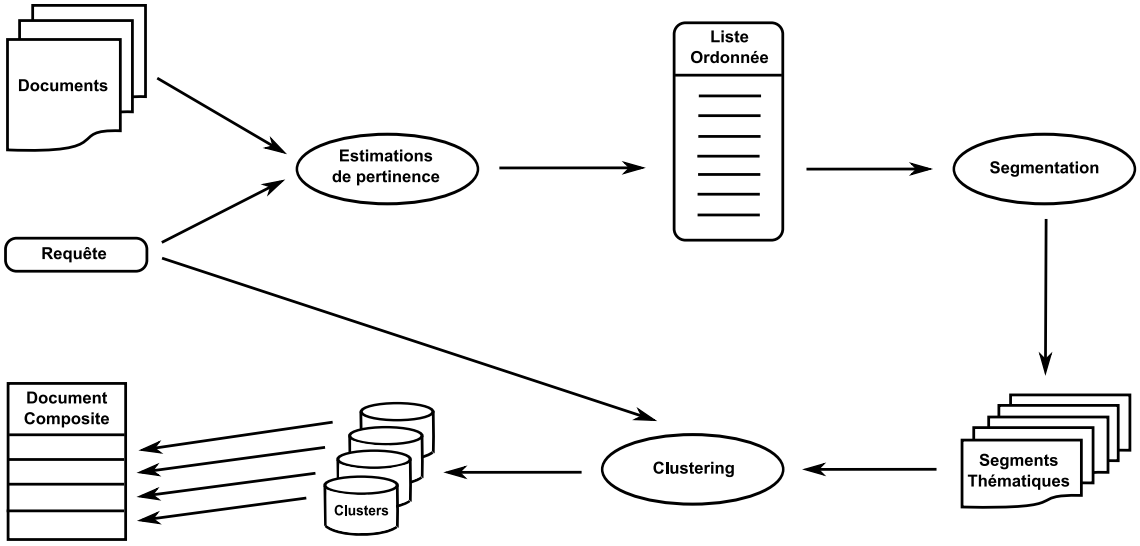

FIGURE 1 – Composition de documents

La problématique rejoint alors celle du résumé multi-documents [Mckeown *et al.*, 1999; Dang, 2005], dont l'objet est de synthétiser les informations contenues dans une collection de documents, et plus particulièrement celle des systèmes de résumés orientés requête [Goldstein *et al.*, 2000; Liu *et al.*, 2006] qui cherchent à produire un texte reprenant les informations principales qu'un utilisateur pourra trouver dans un corpus en rapport avec sa requête. Cependant, le but de notre approche est moins de fournir à l'utilisateur un document contenant l'ensemble des informations répondant à ses besoins, ce qui semble difficile au regard de l'aspect subjectif de la recherche d'information, que de l'aider à orienter sa recherche en lui fournissant un aperçu des différentes thématiques se rapportant à son sujet. Ici, la question n'est pas : comment synthétiser l'information pertinente ? Mais plutôt : comment permettre à un utilisateur d'appréhender rapidement les principales thématiques en rapport avec sa requête ? L'objectif est de fournir à l'utilisateur une sorte de sommaire dont les points d'entrée le conduiront vers l'information qui l'intéresse.

Principales Contributions

Outre la conception du document composite final qui est l'objectif central de cette thèse, les contributions réalisées sont multiples. Elles concernent le découpage des documents et son évaluation, les mesures de pertinence et de similarité des textes, l'im-

pact que peut avoir l'individualisation des thématiques des documents en recherche d'information (et notamment sur les catégories de résultats présentées à l'utilisateur), le mode d'évaluation des systèmes utilisant un clustering des résultats et enfin, la prise en considération du contexte dans les processus de clustering.

Segmentation thématique

Cherchant à individualiser les thématiques des documents, nous nous sommes intéressés dans un premier temps aux processus de segmentation thématique qui visent à découper les documents en passages thématiquement homogènes [Salton *et al.*, 1996]. Une étude menée sur les méthodes et modèles de segmentation thématique de textes nous a conduit à la conception de deux méthodes de segmentation innovantes qui tentent d'adopter une vision plus globale des textes que les approches existantes. La première de ces méthodes fait évoluer un clustering des phrases des documents en introduisant une notion de proximité dans le texte pour déterminer les segments thématiques [Lamprier *et al.*, 2007a; Lamprier *et al.*, 2008g]. La seconde, quant à elle, tente de maximiser deux fonctions objectifs : la similarité entre les phrases d'un même segment et la dissimilarité entre segments adjacents [Lamprier *et al.*, 2007e; Lamprier *et al.*, 2007d; Lamprier *et al.*, 2008c]. Lors de l'élaboration de ces méthodes, nous nous sommes rendu compte des nombreuses difficultés que pose l'évaluation de la segmentation thématique des textes. Certaines mesures existent mais celles-ci présentent certains biais, notamment pour ce qui est de la comparaison de méthodes ne possédant pas les mêmes caractéristiques (présentant par exemple des sensibilités différentes face aux ruptures thématiques des textes). Cela nous a conduit à la mise en place d'une normalisation de ces mesures selon les fréquences de segmentation et à l'élaboration d'une mesure plus équitable basée sur la stabilité des méthodes face aux transformations des textes plutôt que sur la comparaison de la segmentation obtenue avec une segmentation de référence [Lamprier *et al.*, 2007c]. À la lumière des différentes mesures d'évaluation établies, les performances de nos deux méthodes de segmentation semblent surpasser celles des méthodes existantes sur des corpus de textes généraux [Lamprier *et al.*, 2008f]. Les segments produits paraissent sensiblement plus représentatifs des différentes thématiques abordées par les documents.

Étude des mesures de similarité et normalisation

Dans [Singhal *et al.*, 1996b], des expérimentations ont montré que le nombre de mots des documents a un impact direct sur leur probabilité à être considérés comme pertinents par les différentes mesures existantes. L'estimation de la pertinence des segments thématiques, présentant généralement une plus grande hétérogénéité que les documents en terme de taille, risque alors d'être fortement affectée par ce biais observé. Nous avons alors poussé plus avant l'étude réalisée en analysant l'influence de la taille des textes et de la requête sur l'espérance mathématique du score des estimations de pertinence et de similarité entre textes. Ceci nous a conduit à proposer une normalisation des scores qui puisse permettre de juger de la pertinence de textes ou de la proximité inter-textes de

manière plus équitable [Lamprier *et al.*, 2007b].

Impacts de la segmentation sur les estimations de pertinence

De nombreux travaux ont montré qu'un découpage des documents pouvait permettre d'améliorer l'ordre des résultats retournés par les systèmes de recherche d'information. Cela permet en effet d'individualiser des passages susceptibles d'être noyés dans une masse d'informations sans rapport avec le besoin exprimé par l'utilisateur [Callan, 1994]. Cependant, selon [Kaszkiel and Zobel, 2001], le fait de travailler avec des séquences de mots de longueur arbitraire conduit à l'obtention de meilleurs résultats que l'utilisation de segments thématiques qui sont pourtant censés mieux représenter les différentes notions abordées par un document. Cette observation paradoxale semble due à des biais relatifs aux mesures de similarité utilisées pour juger de la pertinence des textes. La normalisation des mesures de similarité établie dans [Lamprier *et al.*, 2007b], nous a permis de montrer dans [Lamprier *et al.*, 2008e] le meilleur potentiel de l'utilisation de segments thématiques pour juger de la pertinence de documents.

Impacts de la segmentation sur le clustering des documents

En appliquant des techniques de clustering aux segments thématiques des documents plutôt qu'aux documents eux-mêmes, nous avons montré qu'une individualisation des thématiques abordées par les documents retournés par une recherche préliminaire pouvait conduire à l'obtention de groupes de résultats représentant mieux les différentes notions abordées [Lamprier *et al.*, 2008h; Lamprier *et al.*, 2008a; Lamprier *et al.*, 2008d] que lorsque les documents sont pris en compte de manière globale. Les clusters présentés à l'utilisateur, puisque mieux centrés sur des sujets spécifiques, sont alors plus facilement interprétables et permettent ainsi une meilleure localisation des informations pertinentes.

Évaluation des systèmes produisant un clustering des résultats

Les systèmes de recherche d'information réalisant une catégorisation de leurs résultats sont généralement évalués selon leur capacité à produire un groupe possédant un fort ratio d'éléments pertinents [Jardine and Van Rijsbergen, 1971]. Considérant que le regroupement de l'information pertinente dans un cluster unique n'est pas nécessairement le meilleur moyen d'aider l'utilisateur dans sa recherche d'information puisque cela ne permet pas de faire émerger les différents aspects de sa requête, nous cherchons à adopter une méthodologie d'évaluation des systèmes qui, plutôt que de ne s'intéresser qu'au meilleur des clusters proposés, permette de rendre compte de la capacité d'accès à l'information pertinente à partir de la liste de descriptions de clusters présentée à l'utilisateur. Cer-

taines approches d'évaluation existantes proposent de reconstruire des listes ordonnées de résultats à partir des groupes formés afin d'évaluer les systèmes utilisant un clustering des résultats de la même manière que les systèmes classiques présentant une simple liste ordonnée à l'utilisateur [Bellot and El-Bèze, 1999; Leuski, 2001b], ce qui permet d'estimer les performances des systèmes sur l'ensemble des groupes qu'ils produisent plutôt que sur un unique cluster tel que c'est généralement le cas. Nous avons néanmoins observé que les approches de reconstruction de listes existantes présentaient une forte tendance à favoriser les systèmes regroupant l'ensemble des textes pertinents dans un même cluster, au détriment des systèmes réalisant une distribution de l'information pertinente sur l'ensemble des groupes afin de proposer des clusters représentatifs des différents aspects de la requête de l'utilisateur. Nous avons alors cherché à établir des approches de reconstruction de listes qui rendent réellement compte de la capacité à atteindre les informations pertinentes à partir de la liste de descriptions de clusters présentée à l'utilisateur [Lamprier *et al.*, 2009]. Ces mesures permettent une comparaison plus équitable entre des systèmes produisant des clusterings de différentes natures.

Prise en compte de la requête dans le processus de clustering

Un certain nombre d'approches ont tenté d'introduire une prise en compte du contexte dans leur processus de clustering [Iwayama, 2000; Tombros *et al.*, 2003; Tombros and Van Rijsbergen, 2004]. Alors qu'avec les mesures d'évaluation classiques, ces approches ne semblent pas apporter d'amélioration significative des résultats, les mesures d'évaluation que nous avons proposées [Lamprier *et al.*, 2009] ont permis de mettre en lumière le fait qu'une considération de la requête dans les calculs de similarité inter-textes pouvait permettre, de manière indirecte, de distinguer certains aspects du besoin exprimé par l'utilisateur et ainsi, de faciliter l'accès à l'information pertinente. Par conséquent, nous avons proposé d'agir directement sur le processus de clustering en utilisant un algorithme génétique dédié aux problèmes multi-objectifs [Zitzler, 1999], afin de considérer conjointement deux critères de cohésion des groupes et de proximité des centres des clusters à la requête [Lamprier *et al.*, 2008b]. En imposant aux groupes de documents de s'organiser autour du sujet concerné par la recherche d'information, nous permettons une distinction plus fine entre les différents types de documents fortement connectés au besoin exprimé. Les résultats obtenus indiquent qu'un tel processus de présentation des résultats permet d'améliorer significativement les capacités de localisation des informations pertinentes. Une détermination automatique du nombre de groupes à produire est intégrée à l'algorithme proposé.

Composition de documents

Le système de composition est calqué sur notre système de clustering de docu-

ments orienté requête, à ceci près que les éléments à considérer sont des segments thématiques plutôt que les documents eux-mêmes. Nous recherchons, à l'aide d'un algorithme multi-objectifs, le sous-ensemble de segments maximisant à la fois un critère de proximité à la requête et un critère de représentativité des thématiques abordées par les documents considérés. Le sous-ensemble de segments ainsi construit constitue notre document composite. Des expériences menées ont montré que l'utilisation de segments plutôt que de documents permet d'améliorer l'organisation des groupes autour de la requête. De plus, le fait de présenter une liste de segments à l'utilisateur plutôt qu'une liste de documents paraît lui permettre d'atteindre l'information pertinente plus facilement [Lamprier *et al.*, 2008b]. D'une manière générale, le système mis en place semble permettre une diminution significative des efforts à fournir pour atteindre les informations recherchées.

Organisation

Cette thèse s'articule autour de trois parties. La première, qui constitue un état de l'art de la recherche d'information, pose les bases du discours mené dans l'ensemble de ce rapport de thèse, permettant de situer les approches que nous proposons dans la multitude de travaux réalisés dans le domaine de la recherche d'information, ainsi que de nous fournir les outils techniques et théoriques nécessaires à leur conception :

- Dans un premier chapitre nous réalisons un survol des principales composantes des systèmes de recherche d'information ;
- Dans un second, nous nous focalisons sur les systèmes réalisant une catégorisation de leurs résultats ;
- Un troisième chapitre s'intéresse à l'évaluation des systèmes de recherche d'information ;
- Puisque nous sommes amenés, à plusieurs reprises, à utiliser des techniques propres au domaine de l'optimisation combinatoire, un quatrième et dernier chapitre réalise une présentation des meta-heuristiques que nous employons.

Cherchant à proposer des catégories de résultats plus représentatives des différentes notions abordées par les documents retournés par une recherche préliminaire, la seconde partie s'intéresse à l'individualisation des thématiques des documents :

- Un premier chapitre se focalise alors sur la segmentation thématique des documents ;
- Un second s'intéresse à l'impact que peut avoir la prise en compte des segments thématiques sur les estimations de pertinence des documents ;
- Un dernier chapitre explore différentes pistes pour considérer ces segments lors de la production de groupes de résultats.

La dernière partie concerne l'organisation des catégories de résultats et la production du document composite final :

- Dans un premier chapitre, nous nous interrogeons sur le type d'organisation

des clusters permettant la meilleure localisation de l'information pertinente. Vaut-il mieux, à l'instar de la plupart des systèmes, chercher à concentrer l'information pertinente dans un unique cluster, ou bien plutôt tenter de distribuer l'information pertinente sur l'ensemble des groupes afin d'en faire émerger les différents aspects ? Nous proposons un certain nombre de mesures d'évaluation permettant la comparaison de ces deux différents points de vue ;

– Dans un dernier chapitre, nous proposons d'agir directement sur le processus de clustering afin d'organiser les différents clusters autour du besoin exprimé par l'utilisateur et ainsi obtenir une liste de représentants de clusters constituant un réel aperçu de l'information pertinente. Lorsqu'appliqué au niveau des segments thématiques des documents, l'algorithme établi représente un processus de recherche des extraits de texte les plus représentatifs des différents aspects de la requête de l'utilisateur.

Enfin, une conclusion générale tente de rapprocher le document composite obtenu des résumés multi-documents orientés requête, puis propose quelques perspectives de travail.

Première partie

État de l'art sur la recherche d'information

La recherche d'information, dont la tâche principale est la conception d'outils d'aide à la localisation d'informations, occupe une place très importante de la recherche en informatique depuis de très nombreuses années ("depuis que les ordinateurs sont capables de compter des mots" [Belew and Van Rijsbergen, 2000]). Nous en présentons ici les approches et concepts fondamentaux.

Alors que le premier chapitre passe en revue les différentes composantes des systèmes de recherche d'information classiques, de l'interprétation des besoins de l'utilisateur au mode de présentation des résultats en passant par la représentation interne des concepts abordés par les documents manipulés, le second chapitre se focalise sur des systèmes plus spécifiques, qui réalisent une catégorisation des résultats présentés à l'utilisateur pour l'orienter plus efficacement vers les informations pertinentes. Bien que, tel que nous le verrons, cette notion de pertinence des informations est une notion très subjective qu'il est difficile de définir de manière précise, le troisième chapitre s'intéresse à l'expérimentation des systèmes de recherche d'information et plus précisément, à l'évaluation de leur capacité à diriger l'utilisateur vers des informations pertinentes. Enfin, étant amenés, à plusieurs reprises, à utiliser des techniques propres au domaine de l'optimisation combinatoire, un quatrième et dernier chapitre réalise une présentation rapide des meta-heuristiques que nous employons, ainsi que des différentes propositions qui ont pu avoir été faites pour leur application dans le domaine de la recherche d'information.

Chapitre 1

Les systèmes de recherche d'information

Ce chapitre présente les différentes composantes des systèmes de recherche d'information classiques. Dans une première section, nous présentons la structure générale d'un système de recherche d'information et nous tentons de cadrer notre discours parmi la variété de champs d'application que le domaine de la recherche d'information couvre. Nous décrivons ensuite les concepts et procédés généralement empruntés par les systèmes pour représenter les documents et les mettre en relation avec les besoins de l'utilisateur. Les modes de formulation de requêtes et de présentation des résultats sont enfin explorés dans les deux dernières sections.

Sommaire

1.1 Présentation et objectifs 12
1.2 Représentation des textes 13
1.2.1 Lemmatisation 14
1.2.2 Sélection des termes représentatifs 15
1.2.3 Structure d'indexation 18
1.3 Estimation de la pertinence des documents 19
1.3.1 Modèles de recherche 19
1.3.2 Mesures de similarité 31
1.4 Formulation d'une recherche d'information 36
1.4.1 Types de requêtes 37
1.4.2 Expansion de requêtes 38
1.5 Présentation des résultats 41
1.6 Conclusion 45

1.1 Présentation et objectifs

L'objectif premier d'un système de recherche d'information est d'aider un utilisateur à localiser les documents contenant les informations qui correspondent à ses besoins. Alors que les premiers systèmes ont été imaginés pour permettre aux visiteurs d'une bibliothèque d'accéder à ses différents ouvrages [Tombros, 2002], les besoins ont depuis bien évolué et la recherche d'information ne se limite plus à la prospection de matériel littéraire. Le domaine a étendu son champ d'action à la recherche de divers types d'informations (e.g. textes, images, audio, vidéo, etc...) dans divers types de corpus (réseaux ou bases de données, documents structurés ou non structurés, existence ou non de liens hypertextes, etc...). L'ensemble des observations réalisées et des approches proposées dans cette thèse concernent la recherche d'information textuelle contenue dans des documents non structurés ne possédant pas de liens hypertextes. L'objet de ce chapitre n'étant pas de réaliser un état de l'art exhaustif sur les systèmes de recherche d'information mais de présenter les concepts et approches utiles à notre discours, nous ne nous attarderons pas sur les systèmes de recherche autres que textuels ou utilisant, outre leur contenu, les caractéristiques des documents tels que leur structure (*e.g.*, documents *XML* [Fuhr *et al.*, 2006]) ou leurs relations explicites (*e.g.*, liens hypertextes [Picarougne, 2004]) pour retourner l'information pertinente. Le lecteur intéressé pourra se référer aux travaux de Baeza-Yates & Ribeiro-Neto [Baeza-Yates and Ribeiro-Neto, 1999], de Belew & Van Rijsbergen [Belew and Van Rijsbergen, 2000] ou de Manning, Raghavan & Schütze [Manning *et al.*, 2008] qui en donnent un bon aperçu. Par ailleurs, nombre des concepts présentés dans la suite de ce chapitre sont communs à l'ensemble des systèmes de recherche d'information, la structure générale restant plus ou moins la même quel que soit le type d'information recherchée ou de corpus interrogé.

D'après la définition donnée par Lancaster [Lancaster, 1968], un système de recherche d'information ne modifie pas les connaissances d'un utilisateur sur un sujet donné : son but est uniquement de l'informer de l'existence et de la localisation d'informations liées à ses besoins dans une collection de documents. Au sens strict, cette définition exclut un certain nombre de branches de la recherche d'information actuelle, notamment les systèmes de Questions-Réponses [Winograd, 1972; Ruthven *et al.*, 2001; Liljenback, 2007], qui visent à répondre précisément à une question posée par un utilisateur, et les systèmes de résumé multi-documents orienté requête [Mani, 2001; Chaâr *et al.*, 2002; Wei *et al.*, 2008], qui cherchent à synthétiser l'information en relation avec le besoin exprimé par l'utilisateur. Néanmoins, elle permet de distinguer les trois composantes majeures d'un système de recherche d'information classique : un utilisateur avec un besoin d'information (exprimé par une requête), une collection de documents dans laquelle l'information est recherchée et enfin, les réponses fournies à l'utilisateur.

La figure 1.1 décrit le fonctionnement général d'un système de recherche d'information classique. On y retrouve les différentes étapes d'une recherche d'information :

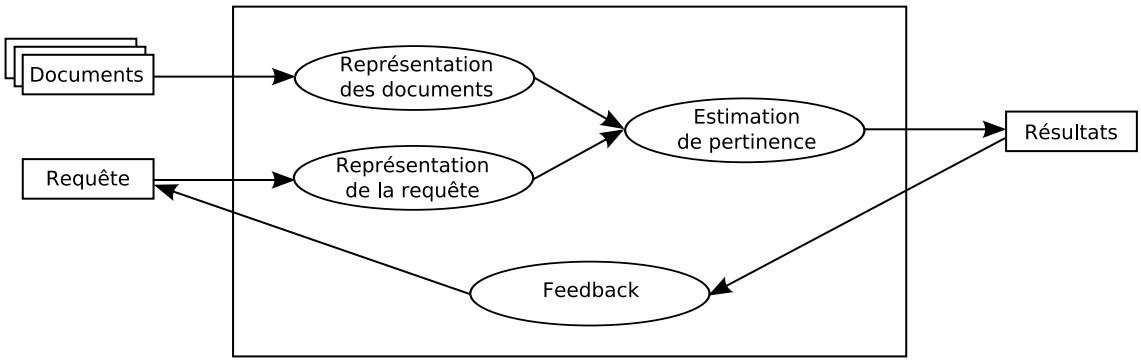

FIGURE 1.1 – Fonctionnement général d'un système de recherche d'information.

expression d'un besoin, représentation des documents et de la requête sous une forme interprétable par le système, mise en correspondance des documents avec le besoin exprimé et enfin, présentation des résultats. Certaines approches y ajoutent un retour de pertinence (relevance feedback) [Salton and Buckley, 1997] permettant au système d'adapter les résultats en fonction d'appréciations faites par l'utilisateur. Ces différentes étapes sont explorées dans la suite de ce chapitre.

1.2 Représentation des textes

L'indexation des documents de la collection sur laquelle porteront les recherches futures constitue la première étape de tout système de recherche d'information. Cette indexation des documents, généralement réalisée une seule fois pour toutes les requêtes, permet la construction de structures dans lesquelles les documents sont représentés afin d'en faciliter l'accès ultérieur. En toute généralité, un index est une liste de descripteurs à chacun desquels est associée une liste des documents (ou de passages de documents) auxquels ce descripteur renvoie. Lors d'une recherche d'information, cet index est utilisé pour rapprocher la requête utilisateur des différents documents du corpus et établir une liste de réponses. L'indexation de documents implique de choisir un mode de représentation qui puisse être interprétable par le système, c'est à dire qui puisse lui permettre d'estimer la proximité thématique des documents avec la requête. Représenter un texte est une tâche difficile car il s'agit d'en formaliser les principaux concepts. Étant donné le nombre de documents à traiter (souvent plus de 100 000 documents), procéder à une analyse sémantique pour déterminer les concepts du texte (telle que celle réalisée dans [Pinon *et al.*, 1997] par exemple) peut paraître difficile. En pratique, on cherche plutôt des représentants des concepts (ou descripteurs). Ceux-ci peuvent être de formes différentes, l'objectif étant de trouver un mode de description qui allie facilité de traitement et précision de représentation du sens. La quasi totalité des systèmes de recherche d'informa-

tion actuels se basent sur le principe que Luhn a établi dans [Luhn, 1958] : "le contenu textuel d'un document discrimine le type et la valeur des informations qu'il véhicule". L'analyse de la présence de mots peut donc permettre de déterminer les documents susceptibles de répondre aux souhaits d'un utilisateur [Van Rijsbergen, 1979]. Dans leur forme la plus simple, les descripteurs utilisés peuvent alors être les différents mots du texte à représenter. Une forme plus complexe de représentation est par exemple celle de [Lewis and Jones, 1996] où des techniques linguistiques sont utilisées pour extraire des mots composés. Par convention, l'unité choisie pour décrire les documents est appelée "terme", que l'on parle de mots simples, de mots composés, de doublets de mots, de triplets, etc...

La suite de cette section concerne la description d'un processus de prétraitement (dit de lemmatisation) généralement appliqué aux textes, la présentation de différents modes de sélection des termes représentatifs et enfin, la définition de la structure la plus souvent employée pour indexer les documents.

1.2.1 Lemmatisation

Dans la langue naturelle, de nombreux mots peuvent se décliner sous différentes formes en fonction de leur genre, de leur nombre, de leur mode, etc... Pour une même racine (lemme), il peut exister une multitude d'écritures. Par exemple, l'adjectif "petit" peut apparaître sous quatre formes : "petit", "petite", "petits" et "petites". Or, la plupart du temps, les différentes formes fléchies d'un mot (conjugaisons, déclinaisons, ...) font référence à des concepts fortement similaires, voire identiques. Il semble alors naturel qu'elles puissent faire référence à une même entrée de l'index. En effet, il est probable que les documents contenant par exemple le mot "généalogique" puissent intéresser un utilisateur ayant effectué une recherche contenant le mot "généalogie". Un processus de lemmatisation, qui consiste à extraire la forme canonique des mots, est alors souvent appliqué aux textes pour améliorer la recherche documentaire en permettant une prise en compte commune des différentes variantes d'un même terme. Par ailleurs, ce processus permet de réduire la taille de l'index et donc d'améliorer les performances des systèmes.

Le *Stemming* est une technique permettant la généralisation des mots suivant leurs variantes morphologiques par élimination ou remplacement de suffixes selon des règles prédéfinies. Par exemple, en français, une règle peut envisager de supprimer le suffixe "ement" de tous les mots du texte ("lentement" devient alors "lent"), une autre peut amener à remplacer tous les suffixes "ation" par "er" ("catégorisation" devient alors "catégoriser") [1]. Des règles de transformation du singulier en pluriel ou du féminin en masculin y sont généralement ajoutées. L'objectif est de faire se référer à la même entrée de l'index tous les mots en relation avec le même concept et affecter à des entrées différentes tous ceux qui ont des sens éloignés. De par sa

1. La mise en place de telles règles de transformation est une tâche très délicate, tant la langue peut comporter des exceptions auxquelles une attention particulière doit être portée (par exemple, la règle de transformation des suffixes "ation" par "er" conduit le mot "fabrication" à se réduire à "fabricer").

simplicité et ses performances observées en recherche d'information, l'algorithme de *Stemming* pour la langue anglaise le plus utilisé dans la littérature est sans aucun doute celui proposé par Porter dans [Porter, 1980].

Néanmoins, l'utilisation de ce genre de techniques de généralisation des mots (mise sous forme canonique) risque d'entraîner quelques transformations abusives [Riloff, 1995] (par exemple "ration" peut devenir "rer" en appliquant notre règle de remplacement des suffixes "ation" en "er"). Afin d'éviter ce genre de problèmes, il est possible d'utiliser un dictionnaire électronique pour savoir si une séquence de lettres en fin de mot correspond bien au suffixe qui s'applique à la règle concernée. Dans ce cas, le mot "ration" est reconnu comme un nom commun à part entière et la règle de remplacement du suffixe "ation" ne s'applique pas. Cette approche a été notamment étudiée pour la langue française dans [Savoy, 1993].

Reste que deux mots écrits de la même manière peuvent ne pas correspondre à la même racine. Par exemple, le mot français "portes" peut être un nom féminin pluriel ou un verbe du premier groupe conjugué à la seconde personne du singulier. Pour améliorer les performances de la recherche d'information, il est utile de distinguer les deux interprétations pour les faire se référer chacune à une entrée de l'index différente. Pour ce faire, il est nécessaire de recourir à une analyse plus profonde du texte. Un étiquetage syntaxique, tel que réalisé dans [El-Bèze and Spriet, 1995] ou [Buvet *et al.*, 2003], peut permettre de distinguer le nom commun du verbe et donc de les transformer de manières différentes pour obtenir deux entrées d'index distinctes.

Certaines approches vont plus loin dans la désambiguisation des termes pour la recherche d'information en réalisant un apprentissage de règles d'inférence [Yarowsky, 1995] ou en utilisant des ressources linguistiques (par exemple un thésaurus [Voorhees, 1993]). Cela permet notamment d'associer les différents synonymes d'un mot à la même entrée de l'index. Au regard du polymorphisme de la langue, une désambiguisation totale est cependant très difficile à réaliser [Krovetz and Croft, 1992; Sanderson, 1997; Stokoe *et al.*, 2003].

1.2.2 Sélection des termes représentatifs

En admettant qu'un terme qui apparaît souvent dans un texte représente un concept important de celui-ci, une première approche de sélection des termes porteurs de sens pourrait consister à choisir une fréquence d'occurrence minimale : si le nombre d'occurrences d'un terme dans un document dépasse un certain seuil, il est considéré comme un bon descripteur des concepts du document concerné et doit donc être utilisé pour son indexation. Cependant, il apparaît que les mots les plus fréquents dans la langue naturelle sont des mots fonctionnels (mots vides). Par exemple, en français les mots les plus fréquents sont "un", "de", "les", etc... La figure 1.2 illustre deux tendances majeures observées par analyse statistique de textes [Losee, 2001] :

– La fréquence d'un terme selon son rang (où les termes sont ordonnés selon leur

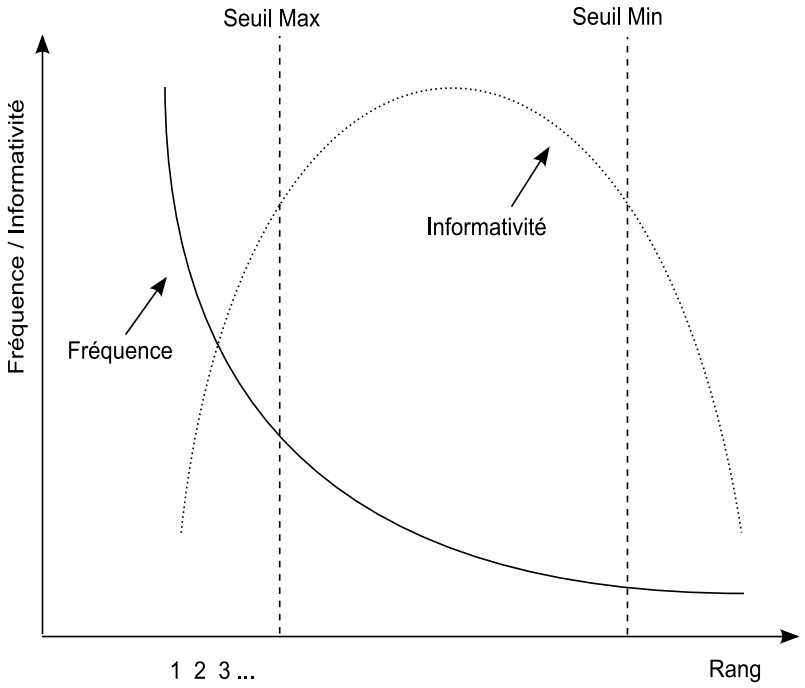

FIGURE 1.2 – Correspondance entre informativité et fréquence des termes

nombre d'occurrences) forme une courbe en hyperbole. Cette courbe suit la loi de Zipf qui établit que le produit de la fréquence d'un terme par son rang est approximativement constant [Zipf, 1949].

– La valeur informative des termes [2] peut s'exprimer sous la forme d'une gaussienne en fonction du rang de leur nombre d'occurrences [Van Rijsbergen, 1979].

Luhn [Luhn, 1958] a utilisé ces observations comme hypothèse de base pour spécifier deux seuils de coupure (seuil max et seuil min) pour déterminer les termes les plus informatifs des documents. Les termes au-delà du seuil maximum sont considérés comme trop communs, ce sont généralement des mots de liaison ou d'usage courant qui ne véhiculent pas ou peu d'information sémantique. Une requête employant uniquement de tels termes serait susceptible de retourner la totalité du corpus s'ils étaient considérés [Salton and McGill, 1986]. Au contraire, les termes en deçà du seuil minimum sont considérés comme trop rares, ils n'apportent que rarement des informations nécessaires à la compréhension d'un texte. Il n'existe pas de méthode "optimale" pour établir ces seuils, la meilleure méthode semblant être de procéder par essais successifs sur le corpus concerné [Van Rijsbergen, 1979]. Dans

2. La valeur informative des termes n'étant pas une notion définie très précisément en recherche d'information, elle est utilisée de façon intuitive dans la plupart des études. On peut cependant trouver son équivalent en théorie de l'information [Shannon, 1948].

le cas général, choisir pour termes descripteurs ceux dont le nombre d'occurrences appartient à l'intervalle $\left[\frac{|\mathcal{M}|}{100}, \frac{|\mathcal{M}|}{10}\right]$, avec $|\mathcal{M}|$ le nombre de mots dans le corpus, permet de représenter les textes selon leurs termes les plus informatifs [Salton *et al.*, 1974].

Plutôt que d'utiliser ce genre de seuils pour sélectionner les termes importants, la plupart des modèles de recherche d'information (section 1.3), notamment le modèle vectoriel [Baeza-Yates and Ribeiro-Neto, 1999] qui est au centre de nos expérimentations, réalisent une pondération des termes selon leur fréquence dans le document concerné (qui traduit leur importance pour décrire les concepts du document) et leur répartition dans le corpus (qui traduit leur pouvoir discriminant) pour permettre une prise en compte de leur caractère informatif. Différentes formulations de la pondération des termes sont proposées dans la littérature [Salton and Buckley, 1988; Harman, 1992b; Kwok, 1996; Zobel and Moffat, 1998]. La plus connue et la plus couramment utilisée à ce jour est certainement la formulation en $tf * idf$ [Salton and Buckley, 1988] qui regroupe un ensemble de schémas de pondération et de sélection de termes [3]. Dans cette formulation, le poids $w_{D_i,t}$ de chaque terme t dans chaque document D_i correspond au produit de deux composantes principales, la fréquence $tf_{D_i,t}$ qui dépend du nombre d'occurrences du terme t dans le document D_i et la composante idf_t qui diminue lorsque le nombre de documents contenant le terme t augmente (correspond au pouvoir discriminant du terme) [Jones, 1988]. On retrouve dans la littérature plusieurs formules tf et idf. La formulation préconisée dans [Buckley *et al.*, 1994], que nous utilisons par la suite, est donnée par :

$$tf_{D_i,t} = 1 + \ln\left(Nb_{t \in D_i}\right) \tag{1.1}$$

$$idf_t = \ln\left(\frac{Nb_D}{Nb_{D \ni t}}\right) \tag{1.2}$$

avec t un terme du texte D_i, $Nb_{t \in D_i}$ le nombre d'occurrences de ce terme t dans le texte D_i, Nb_D le nombre total de textes contenus par le corpus étudié et $Nb_{D \ni t}$ le nombre de textes contenant ce terme t. En fixant un poids $w_{D,t}$ minimal (avec $w_{D_i,t} = tf_{D_i,t} \times idf_t$), il est possible de sélectionner les termes les plus représentatifs en considérant à la fois leur pouvoir discriminant et leur importance dans le texte concerné. Par ailleurs, de nombreux modèles utilisent ce genre de pondération dans leurs estimations de pertinence [Zobel and Moffat, 1998] (voir section 1.3). Le filtrage des termes n'est alors plus primordial puisque les termes les moins représentatifs ne pèsent que très peu dans la recherche des textes en relation avec la requête de l'utilisateur.

Néanmoins, certains mots fonctionnels, tels que "quant" ou "auparavant", n'apparaissent que rarement dans les textes. Il n'est alors pas possible de les éliminer par considération de leur fréquence d'occurrence. Afin d'éliminer ces mots malgré tout, de nombreux systèmes utilisent une liste, appelée *stop-list* qui contient l'ensemble

3. tf signifie "term frequency" et idf signifie "inverted document frequency"

des mots vides que l'on cherche à éviter. Le pouvoir discriminant des mots pouvant varier selon le domaine d'application du système, de nombreuses listes de mots vides existent. Néanmoins, la liste de mots vides proposée dans [Fox, 1992] est utilisée dans la majorité des systèmes de recherche d'information en langue anglaise [4].

1.2.3 Structure d'indexation

Une fois les termes des documents extraits, le processus d'indexation doit les ranger dans une structure pour permettre l'accès aux données. Une première possibilité pourrait être de créer une base dans laquelle les entrées seraient tout simplement les différents documents du corpus. Dans ce cas, la recherche des documents qui correspondent aux besoins de l'utilisateur se ferait de manière séquentielle : le système explorerait, dans l'ordre où elles ont été créées, chacune des entrées de la base pour déterminer celles qui renvoient à un maximum de termes en relation avec la requête. Néanmoins, si cette technique de recherche peut ne durer que quelques secondes sur un corpus ne contenant que quelques centaines de documents, l'opération peut se révéler extrêmement coûteuse si la base atteint des milliers d'entrées ; d'autant plus si chacune d'entre elles renvoie à un grand nombre de termes. Conserver les données sous cette forme risque de conduire à des recherches très lentes, ne permettant pas d'atteindre l'information recherchée dans un temps raisonnable [5]. Par ailleurs, avec une telle approche, la structure d'indexation risque de nécessiter une très grande place mémoire [Frakes and Baeza-Yates, 1992].

Par conséquent, la majorité des approches renversent le problème en choisissant les termes, plutôt que les documents du corpus, comme entrées du fichier d'index. On désigne ce genre d'index sous le nom de fichier inversé. Un fichier inversé dresse, pour chaque terme, la liste des documents qui le contiennent. Non seulement l'utilisation d'un tel fichier permet de diminuer considérablement les besoins en espace mémoire, mais le temps requis pour effectuer une recherche s'en trouve alors considérablement réduit [Van Rijsbergen, 1979]. En effet, avec ce type de structure, la recherche des documents en relation avec le besoin de l'utilisateur ne nécessite pas le parcours de toute la base puisqu'il suffit d'examiner les documents pointés par les entrées d'index correspondant aux termes de la requête. Les calculs d'estimation de pertinence sont alors grandement facilités.

De nombreux autres types de structures ont été proposés, notamment des structures utilisant des arbres, des tables de hachage ou des listes chaînées [Kugel, 1962; Salton, 1962; Vaishnavi, 1989; Frakes and Baeza-Yates, 1992] ainsi que des structures plus élaborées dédiées à la recherche de passages de documents [Kaszkiel *et al.*, 1999]. Cependant, du fait de ses performances et de sa simplicité, l'indexation en fichier inversé reste l'approche la plus employée.

4. Cette liste est utilisée dans l'ensemble de nos expérimentations.

5. Des études ont montré que les utilisateurs de moteurs de recherche abandonnent en moyenne leur prospection d'informations au bout de douze minutes de recherche infructueuse [Leuski, 2001b].

Afin de permettre à l'utilisateur de spécifier au mieux sa requête, certaines approches ont inclus des opérateurs logiques, de proximité, d'adjacence ou de troncature dans leur structure d'indexation initiale. Par exemple, ces opérateurs peuvent permettre de préciser la position des termes les uns par rapport aux autres en les entourant de guillemets (tel que dans le célèbre moteur de recherche Google [Google, 2008]). Ils peuvent également permettre la recherche de termes syntaxiquement (voire sémantiquement) proches des termes de la requête. Ces types d'opérateurs sont très souvent utilisés par les moteurs de recherche actuels [Picarougne, 2004]. Néanmoins, bien que de tels opérateurs soient susceptibles de présenter un certain intérêt pour les approches que nous proposons par la suite, nous nous sommes limité, dans le cadre de cette thèse, à des structures d'indexation en fichier inversé sans utilisation d'opérateurs additionnels.

1.3 Estimation de la pertinence des documents

Si le processus d'indexation a permis de sélectionner et stocker les termes permettant de représenter le contenu des documents d'un corpus, la caractérisation des documents ne se limite pas à leur référencement dans une base de données. Reste à savoir comment interpréter les données pour répondre au mieux au besoin d'information de l'utilisateur. Afin de fournir à l'utilisateur un ensemble de documents correspondant au sujet exprimé par sa requête, de nombreux modèles de recherche ont été établis. Il s'agit de définir des modes de comparaison entre un document et une requête pour en déterminer le degré de correspondance (ou similarité). Dans cette section, nous exposons les modèles de recherche les plus fréquemment rencontrés dans la littérature [6] puis nous présentons en détail quelques mesures de similarité que nous employons dans les chapitres suivants de ce rapport.

1.3.1 Modèles de recherche

Le modèle joue un rôle central en recherche d'information. C'est lui qui détermine le comportement des systèmes. Bien que nous ayons décrit le processus d'indexation dans une section distincte, le mode d'estimation de pertinence n'est pas réellement dissociable de la représentation des documents. En effet, alors que nous avons présenté de manière générale les moyens couramment utilisés dans la littérature pour représenter les textes, de nombreuses questions restent en suspens concernant la façon de déterminer les documents les plus proches de la requête d'un utilisateur. Quels types de termes sont utilisés (mots, doublets, triplets, ...) ? Les termes de la requête doivent-ils tous apparaître dans les documents retournés ? Quels sont les termes les plus importants ? Cherchons nous à associer un degré de pertinence aux

6. Cette liste de modèles est bien sûr loin d'être exhaustive. Nous n'abordons notamment pas les modèles connexionnistes [Boughanem and Soulé-Dupuy, 1992], les modèles inférentiels [Nie and Brisebois, 1996] ni les modèles DFR (Divergence from Randomness model) [Amati and Van Rijsbergen, 2002].

documents ou bien cherchons nous simplement à fournir un ensemble de documents potentiellement pertinents ? D'une manière générale, quels moyens mettre en œuvre pour déterminer qu'un document donné correspond mieux qu'un autre aux besoins de l'utilisateur ? Chaque modèle de recherche d'information présente ses propres caractéristiques, réalise ses propres choix pour interpréter les données. Nous décrivons ici brièvement certains de ces modèles par ordre chronologique de mise en œuvre dans les systèmes de recherche d'information.

Modèle "Matching Score"

Peut-être le premier modèle utilisé en recherche d'information, le modèle "Matching Score" est assez intuitif. Dans ce modèle, le degré de similarité entre un document et une requête correspond à la somme des fréquences (nombre d'occurrences) des termes t de la requête R dans le document D :

$$Sim(R, D) = \sum_{t \in R} Nb_{t \in D} \tag{1.3}$$

Ce calcul revient à parcourir le document pour en compter le nombre d'occurrences des termes de la requête. Plus le score est élevé, plus le document en question est considéré correspondre à la requête et donc plus il aura de chances d'être retourné à l'utilisateur. Ce modèle utilise le résultat brut de l'indexation, sans réaliser aucune réorganisation ou pondération d'aucune sorte. Malgré son extrême simplicité, ce modèle n'est quasiment plus utilisé dans les systèmes actuels du fait de ses mauvaises performances [Baeza-Yates and Ribeiro-Neto, 1999].

Modèle booléen

Dès les années 1950, avec l'apparition des premiers systèmes de recherche d'information, le modèle booléen s'est imposé grâce à sa simplicité de mise en place [Salton and McGill, 1986]. C'est un modèle ensembliste qui cherche à établir une correspondance logique entre la requête et les documents du corpus. Dans ce modèle, un document D est considéré comme une conjonction logique ($\wedge$) de n_t termes t_i (non pondérés) :

$$D = t_1 \wedge t_2 \wedge t_3 \wedge \cdots \wedge t_{n_t} \tag{1.4}$$

La requête R peut utiliser d'autres opérateurs tels que la disjonction ($\vee$) ou la négation ($\neg$), par exemple :

$$R = (t_1 \wedge t_2) \vee (t_1 \wedge \neg t_3) \tag{1.5}$$

Un document correspond à une requête, et donc est présenté à l'utilisateur, si et seulement si l'implication $D \Rightarrow R$ est valide. Cette correspondance $C(D, R)$ peut

être déterminée comme suit (avec R_1 et R_2 des décompositions d'une requête R) :

$$\begin{array}{lcl} C(D,t_i) & = & 1\ Si\ t_i \in D;\ 0\ Sinon \\ C(D,R_1 \wedge R_2) & = & 1\ Si\ C(D,R_1)=1\ Et\ C(D,R_2)=1;\ 0\ Sinon \\ C(D,R_1 \vee R_2) & = & 1\ Si\ C(D,R_1)=1\ Ou\ C(D,R_2)=1;\ 0\ Sinon \\ C(D,\neg R) & = & 1\ Si\ C(D,R)=0;\ 0\ Sinon \end{array} \tag{1.6}$$

Ce modèle, qui permet de filtrer relativement efficacement les documents en adéquation avec la requête, présente néanmoins une limite majeure : le résultat de l'estimation de pertinence étant binaire (un document est ou n'est pas en adéquation avec la requête), il n'est alors pas possible de déterminer qu'un document est plus proche du sujet qu'un autre et donc d'ordonner les documents présentés à l'utilisateur par ordre de pertinence estimée. Cela pose problème lorsque de nombreux documents du corpus se trouvent en adéquation avec la requête. Par ailleurs, l'absence de pondération des termes dans ce modèle peut elle aussi poser problème puisque tous les termes d'un document ou d'une requête ont alors la même importance, quels que soient leur fréquence et leur pouvoir discriminant. Ce modèle n'est alors généralement plus utilisé qu'en tant que première étape des systèmes de recherche d'information, pour réaliser un filtrage préliminaire des documents potentiellement pertinents [Picarougne, 2004].

Modèle booléen étendu

Face aux limites énoncées, Salton [Salton *et al.*, 1983] a proposé d'étendre le modèle booléen en y introduisant une prise en compte de l'importance relative des termes et en associant un degré de pertinence aux documents retournés. Plutôt que de considérer l'adéquation d'un document à une requête de manière binaire, l'utilisation d'une pondération des termes (par exemple en $tf * idf$ [7], voir formules 1.1 et 1.2 section 1.2.2) permet de distinguer les documents fortement caractérisés par les termes les plus discriminants de la requête. Suivant le cadre classique des ensembles flous proposé par Zadeh [Zadeh, 1965], il est possible d'utiliser les formules suivantes pour estimer la correspondance d'un document avec une requête (avec w_{D,t_i} le poids du terme t_i dans le document D) :

$$\begin{array}{lcl} C(D,t_i) & = & w_{D,t_i} \\ C(D,R_1 \wedge R_2) & = & \min\left(C(D,R_1), C(D,R_2)\right) \\ C(D,R_1 \vee R_2) & = & \max\left(C(D,R_1), C(D,R_2)\right) \\ C(D,\neg R) & = & 1 - C(D,R) \end{array} \tag{1.7}$$

Avec ce type de formulation, le score obtenu n'est plus 0 ou 1 mais est compris entre ces deux valeurs selon les poids des termes de la requête dans le document

7. Il est à noter que l'utilisation de cette pondération avec le modèle booléen risque de favoriser les documents les plus longs, où le nombre d'occurrences des termes de la requête a plus de chances d'être élevé. Il serait préférable d'utiliser une formule de tf qui soit normalisée selon la taille du document concerné.

concerné. D'un point de vue théorique, cette formulation n'est pas valide puisque $C(D, R \wedge \neg R) \neq 0$ et $C(D, R \vee \neg R) \neq 1$. En pratique cependant, elle permet de construire une liste ordonnée de documents que l'on peut présenter à l'utilisateur. Cette présentation des résultats permet de réduire grandement les efforts à fournir pour trouver des informations pertinentes puisque les documents les plus intéressants sont supposés être positionnés en début de liste. De plus, le pouvoir discriminant des termes et leur importance dans les documents peuvent être pris en compte, ce qui améliore grandement la qualité de l'estimation de pertinence [Salton *et al.*, 1983].

Néanmoins, il est à noter que l'évaluation des opérateurs logiques de conjonction et de disjonction conduit a des pertes d'informations : lors de l'évaluation d'une conjonction, seule la partie de la requête obtenant le plus mauvais score est considérée, le degré d'adéquation de la meilleure partie est alors ignoré. Le problème inverse est observé avec l'évaluation d'une disjonction, seule la partie de la requête obtenant le meilleur score est considérée. Il en résulte que des documents peuvent obtenir des scores identiques d'adéquation avec la requête, malgré de grandes différences sur des fragments de celle-ci. Afin d'éviter cette perte d'information, de nombreuses autres formes de correspondances ont été proposées. Une formulation fréquemment employée utilise les relations introduites par Lukasiewicz des années auparavant [Lukasiewicz, 1963] :

$$\begin{array}{lcl} C(D, t_i) & = & w_{D,t_i} \\ C(D, R_1 \wedge R_2) & = & C(D, R_1) \times C(D, R_2) \\ C(D, R_1 \vee R_2) & = & C(D, R_1) + C(D, R_2) - C(D, R_1) \times C(D, R_2) \\ C(D, \neg R) & = & 1 - C(D, R) \end{array} \tag{1.8}$$

En utilisant cette formulation, les deux parties d'une conjonction ou d'une disjonction contribuent conjointement à l'évaluation de la correspondance du document à la requête, ce qui améliore grandement les résultats. Reste qu'une requête correspondant à une longue conjonction est très difficile à satisfaire et que dans bien des cas, la réponse fournie par le système risque d'être vide (alors que des documents peuvent répondre partiellement aux besoins de l'utilisateur). Dans l'autre extrême, une longue disjonction est très facile à satisfaire et l'on risque d'obtenir un trop grand nombre de réponses. Afin de permettre un contrôle du nombre de réponses fournies tout en conservant les avantages de la précédente formulation, le modèle p-norme introduit par Salton considère des degrés de satisfaction de conjonctions et de disjonctions plutôt que d'interpréter les opérateurs logiques de manière stricte [Salton *et al.*, 1983]. L'établissement de ce modèle découle de l'observation d'une table de vérité (table 1.1). Dans le cas d'une conjonction, la meilleure correspondance est atteinte au niveau de la dernière ligne, lorsque les deux prédicats A et B sont vrais. Dans le cas d'une disjonction, la pire correspondance correspond à la première ligne, lorsque les deux prédicats sont faux. L'idée est alors d'évaluer dans quelle mesure un point défini par les deux parties A et B d'une conjonction est proche du point (1,1) à atteindre et dans quelle mesure un point défini par les deux parties A et B d'une

A	B	$A \wedge B$	$A \vee B$
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	1

TABLE 1.1 – Table de vérité

disjonction est proche du point (0,0) à éviter (voir figure 1.3). Salton propose les

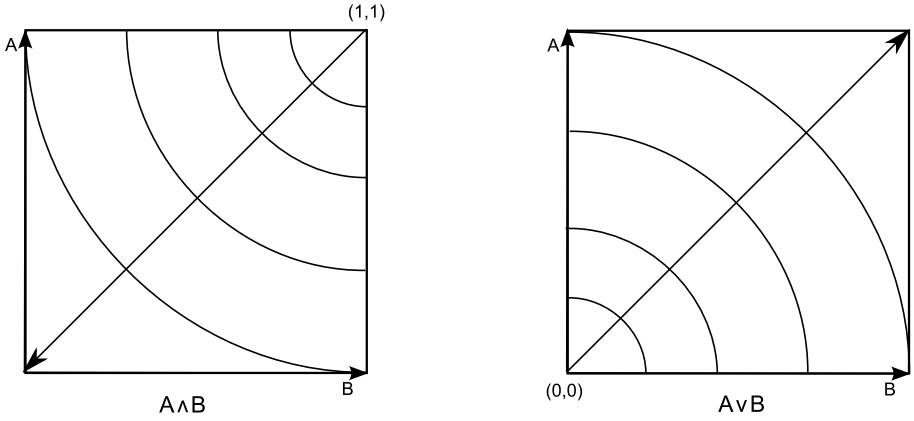

FIGURE 1.3 – Évaluation d'une conjonction et d'une disjonction.

relations de correspondance normalisées suivantes pour réaliser ce calcul de distance [Salton *et al.*, 1983] :

$$\begin{array}{lcl} C(D,t_i) & = & w_{D,t_i} \\ C(D,R_1 \wedge R_2) & = & 1-\sqrt{\frac{(1-C(D,R_1))^2+(1-C(D,R_2))^2}{2}} \\ C(D,R_1 \vee R_2) & = & \sqrt{\frac{C(D,R_1)^2+C(D,R_2)^2}{2}} \\ C(D,\neg R) & = & 1-C(D,R) \end{array} \tag{1.9}$$

Cette formulation permet une évaluation plus souple des opérateurs logiques. Les performances de ce modèle p-norme semblent largement surpasser celles des autres formulations utilisées pour étendre le modèle booléen standard [Fox, 1983]. De nombreuses variantes de ce modèle existent, notamment des formulations permettant une pondération des opérateurs logiques [Hong *et al.*, 2007].

Modèle vectoriel

Plutôt que de chercher une correspondance "logique" entre documents et requêtes, le modèle vectoriel [Salton, 1971b; Baeza-Yates and Ribeiro-Neto, 1999] aborde le problème de la recherche d'information de manière algébrique en s'appuyant sur la théorie mathématique des espaces vectoriels et plus généralement de l'algèbre linéaire. Dans le cadre du modèle vectoriel, les documents et les requêtes sont représentés dans un même espace vectoriel de n_t dimensions (avec n_t le nombre de termes dans le corpus). Ainsi, chaque texte T (document ou requête) est représenté par un vecteur $\vec{T} = (w_{T,t_1}, w_{T,t_2}, \ldots, w_{T,t_{n_t}})$ dont les coordonnées correspondent au poids de chaque terme dans le celui-ci (en utilisant par exemple une pondération en $tf * idf$, voir formules 1.1 et 1.2). Ce vecteur, qualifié dans la littérature de profil lexical, permet de mettre en place des mesures algébriques pour estimer la similarité thématique du texte concerné avec la requête ou les autres textes du corpus.

La figure 1.4 illustre la représentation de deux documents, D_1 et D_2, et d'une requête R dans le modèle vectoriel. Dans cet exemple simplifié, l'espace vectoriel ne comprend que deux dimensions, correspondant à deux termes t_1 et t_2. Les vecteurs $\vec{D_1}$, $\vec{D_2}$ et $\vec{R}$ dépendent des poids de ces deux termes dans les textes correspondants : $\vec{D_1} = (2, 0)$, $\vec{D_2} = (0, 2)$ et $\vec{R} = (2, 1)$. Les deux angles $\theta 1$ et $\theta 2$, entre le vecteur de la requête et ceux des documents D_1 et D_2 respectivement, déterminent le score d'estimation de pertinence des deux documents qui permettra de les classer dans la liste ordonnée présentée à l'utilisateur. Dans cet exemple, le document D_1 sera mieux classé que le document D_2 puisque son vecteur est plus proche de celui de la requête.

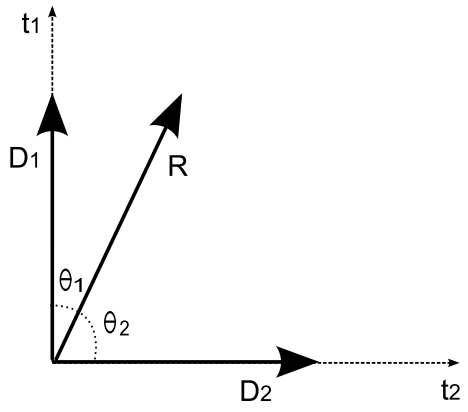

FIGURE 1.4 – Représentation dans le modèle vectoriel.

La mesure Cosine [Salton and McGill, 1986], utilisée dans la plupart des systèmes

s'appuyant sur le modèle vectoriel [8], considère le cosinus de l'angle entre deux vecteurs pour déterminer le degré de similarité thématique des textes correspondants. Elle retourne un score compris entre 0 (aucun terme en commun) et 1 (textes identiques) correspondant à la similarité thématique $Sim(T, T')$ de deux textes T et T' :

$$Sim(T, T') = \cos(\vec{T}, \vec{T'}) = \frac{\sum_{t \in T \cap T'} w_{T,t} \times w_{T',t}}{\sqrt{\sum_{t \in T} w_{T,t}^2 \times \sum_{t \in T'} w_{T',t}^2}} \tag{1.10}$$

Outre l'estimation de la proximité d'un document avec une requête, le modèle vectoriel présente l'avantage non négligeable de permettre des calculs de similarité thématique entre documents du corpus, ce qui n'est pas possible dans le modèle booléen. Ce calcul de similarité entre documents est utile pour de nombreuses tâches, notamment la classification de documents (affectation des documents à des catégories pré-définies), le clustering de documents (détermination automatique de catégories permettant de partitionner au mieux l'ensemble de documents, voir chapitre 2) et à la prise en compte de retours de pertinence (ou *relevance feedbacks*, voir section 1.4.2). De plus, contrairement aux modèles précédents, le modèle vectoriel, et plus particulièrement la mesure Cosine, inclut un facteur de normalisation selon la longueur des textes dans son calcul de similarité : le quotient de la formule 1.10 permet de limiter l'avantage que pourraient avoir les documents contenant un grand nombre de termes sur les plus petits textes [9].

Bien qu'il ait été critiqué à de maintes reprises [10] [Raghavan and Wong, 1985], le modèle vectoriel est apparu au moins aussi performant que de nombreux autres modèles dans de multiples campagnes d'évaluation [Voorhees and Harman, 1997; Manning *et al.*, 2008]. De par sa rapidité de traitement, sa propension à retourner les meilleurs documents en tête de liste et sa capacité à comparer les documents entre eux, le modèle vectoriel est certainement le modèle le plus employé dans la littérature.

Notons enfin que Wong et al. [Wong *et al.*, 1985] ont proposé une généralisation du modèle vectoriel qui tente d'établir un cadre formel dans lequel les dépendances entre les termes peuvent être représentées. Néanmoins, malgré le gain réalisé en terme de qualité des réponses fournies à l'utilisateur, le modèle vectoriel généralisé n'est que peu utilisé en raison de sa grande complexité algorithmique.

8. De nombreuses autres mesures de comparaison de vecteurs existent, notamment la mesure de Jaccard ou la mesure de Dice [Kowalski, 1997], mais la mesure Cosine est la plus répandue.

9. Les techniques de normalisation selon la longueur des textes sont explorées plus en détail en sections 1.3.2 et 6.2.

10. Parmi les principales critiques exprimées figurent notamment une remise en cause de l'hypothèse d'indépendance des termes et la non prise en compte de l'adjacence ou de l'ordre des mots dans le texte.

Modèle probabiliste

L'idée de base du modèle probabiliste, telle que formulée initialement dans [Maron and Kuhns, 1960], est que l'ensemble idéal de documents à retourner à l'utilisateur peut être caractérisé par un sous-ensemble de termes d'indexation. Le processus de recherche probabiliste consiste alors à ordonner les documents du corpus en fonction de la présence, ou de l'absence, des termes possédant une forte probabilité d'être caractéristique de cet ensemble idéal. Le problème est qu'il nous est impossible de savoir *a priori* quels sont les documents pertinents pour une requête donnée, ni combien le corpus en contient. Ces informations doivent être déduites d'une recherche d'information préliminaire ou bien faire appel à des retours de pertinence de la part de l'utilisateur [Van Rijsbergen, 1979]. Le principe du modèle probabiliste est de s'appuyer sur ces connaissances acquises pour estimer des probabilités d'appartenance d'un document donné à l'ensemble des documents pertinents. De nombreux modèles probabilistes existent, nous décrivons ici le plus classique/populaire d'entre eux.

Avec $Pert$ et $NPert$ les ensembles contenant respectivement les documents pertinents et non pertinents à une requête, l'objectif est de définir les probabilités $P(D|Pert)$ et $P(D|NPert)$ que le document D appartienne à l'un et l'autre de ces deux ensembles, pour attribuer un score de pertinence $RSV(D)$ (Retrieval Status Value) au document D [Manning *et al.*, 2008] :

$$RSV(D) = \log \frac{P(D|Pert)}{P(D|NPert)} \tag{1.11}$$

Afin de déterminer ces deux probabilités, un document est généralement décomposé en un ensemble de termes t_i. Ainsi :

$$P(D|Pert) = P(t_1 = x_1, t_2 = x_2, t_3 = x_3, \ldots, t_n = x_n|Pert) \tag{1.12}$$
$$P(D|NPert) = P(t_1 = x_1, t_2 = x_2, t_3 = x_3, \ldots, t_n = x_n|NPert) \tag{1.13}$$

où $ti = xi$ correspond à la présence ou l'absence (1 ou 0) du terme t_i dans le document D [11]. Malgré le fait que les présences ou les absences de termes dans les documents soient des événements dépendants, et qu'en théorie il faudrait tenir compte de ces dépendances, la plupart des approches supposent que les événements liés à différents termes sont indépendants pour simplifier les calculs [Baeza-Yates and Ribeiro-Neto, 1999] [12]. On peut ainsi calculer les probabilités $P(D|Pert)$ et

11. Certaines approches utilisent une pondération des termes dans le modèle probabiliste plutôt que de n'utiliser que des informations binaires de présence ou d'absence des termes dans le document [Fuhr, 1989].

12. La prise en compte des dépendances entre termes implique de disposer d'une base d'échantillons conséquente sur lesquels des calculs complexes doivent être réalisés. Il faut en effet tenir compte de très nombreuses probabilités conditionnelles : $P(t_2 = x_2|t_1 = x_1, Pert)$, $P(t_3 = x_3|t_1 = x_1, t_2 = x_2, Pert)$, $P(t_4 = x_4|t_1 = x_1, t_2 = x_2, t_3 = x_3, Pert)$, etc ... Néanmoins, de nombreuses approches se sont risqué à une telle tâche, notamment par utilisation d'arbres de dépendance [Van Rijsbergen, 1979; Friedman and Goldszmidt, 1996] ou de réseaux Bayésiens [Turtle and Croft, 1989; Callan *et al.*, 1995] et ont rencontré un certain succès [Cho *et al.*, 2003; Manning *et al.*, 2008].

$P(D|NPert)$ de la manière suivante :

$$\begin{aligned} P(D|Pert) &= \prod_{i=1}^{n} P(t_i = x_i|Pert) \\ &= \prod_{i=1}^{n} P(t_i = 1|Pert)^{x_i} P(t_i = 0|Pert)^{(1-x_i)} \qquad (1.14) \\ P(D|NPert) &= \prod_{i=1}^{n} P(t_i = x_i|NPert) \\ &= \prod_{i=1}^{n} P(t_i = 1|NPert)^{x_i} P(t_i = 0|NPert)^{(1-x_i)} \qquad (1.15) \end{aligned}$$

Pour simplifier, on pose $p_i = P(t_i = 1|Pert)$, $q_i = P(t_i = 1|NPert)$, $(1 - p_i) = P(t_i = 0|Pert)$ et $(1 - q_i) = P(t_i = 0|NPert)$. Le score $RSV(D)$ obtenu par le document devient alors :

$$\begin{aligned} RSV(D) &= \log \frac{\prod_{i=1}^{n} p_i^{x_i}(1-p_i)^{(1-x_i)}}{\prod_{i=1}^{n} q_i^{x_i}(1-q_i)^{(1-x_i)}} \\ &= \sum_{i=1}^{n} [\log p_i^{x_i} + \log(1-p_i)^{(1-x_i)} - \log q_i^{x_i} - \log(1-q_i)^{(1-x_i)}] \\ &= \sum_{i=1}^{n} x_i [\log \frac{p_i(1-q_i)}{q_i(1-p_i)}] + \sum_{i=1}^{n} \log \frac{(1-p_i)}{(1-q_i)} \qquad (1.16) \end{aligned}$$

On peut noter que le document n'intervient plus dans la deuxième partie de cette formule (aucun x_i). C'est une constante pour tous les documents, elle peut alors être retirée du calcul. On a alors :

$$RSV(D) = \sum_{i=1}^{n} x_i [\log \frac{p_i(1-q_i)}{q_i(1-p_i)}] \qquad (1.17)$$

La question est alors réduite à l'estimation des probabilités p_i et q_i. Pour ce faire, il faut disposer d'un ensemble d'échantillons de documents pour lesquels des jugements (ou estimations) de pertinence ont été réalisés. En s'appuyant sur cette base de connaissances, la table de distribution 1.2 peut être construite pour chaque terme t_i.

Les estimations des probabilités p_i et q_i découlent directement des observations réalisées :

$$\begin{aligned} p_i &= \frac{S_i}{T_i} \qquad (1.18) \\ q_i &= \frac{P - S_i}{N - T_i} \qquad (1.19) \end{aligned}$$

Documents	Pertinents	Non Pertinents	Total
Terme t_i présent : $x_i = 1$	S_i	$T_i - S_i$	T_i
Terme t_i absent : $x_i = 0$	$P - S_i$	$(N - P) - (T_i - S_i)$	$N - T_i$
Total	P	$N - P$	N

TABLE 1.2 – Présence du terme vs. Pertinence du document

En remplaçant les p_i et q_i par les probabilités correspondantes dans la formule 1.17, on peut obtenir la formule finale d'estimation de pertinence d'un document :

$$RSV(D) = \sum_{i=1}^{n} x_i [\log \frac{S_i/(T_i - S_i)}{(S_i - P)/(N - P - T_i + S_i)}] \tag{1.20}$$

La formule de Robertson-Sparck Jones [Robertson and Jones, 1988], qui est un lissage (*smoothing*) de cette dernière formulation permettant notamment d'éviter des divisions par zéro (par exemple lorsque tous les documents pertinents identifiés possèdent un terme donné), est souvent utilisée dans les approches probabilistes de recherche d'information :

$$RSV(D) = \sum_{i=1}^{n} x_i [\log \frac{(S_i + 0.5)/(T_i - S_i + 0.5)}{(S_i - P + 0.5)/(N - P - T_i + S_i + 0.5)}] \tag{1.21}$$

Nombre de mesures de similarité performantes, notamment la mesure BM25 (ou mesure Okapi à cause du nom du premier système à l'avoir implémentée) [Robertson *et al.*, 1992] et la mesure Inquery [Callan *et al.*, 1995] que nous étudions plus avant en section 1.3.2, sont dérivées de ce modèle pour estimer de la pertinence des documents. Outre ces estimations de pertinence, les modèles probabilistes sont parfois employés pour la sélection des termes d'indexation [Robertson *et al.*, 1981].

Modèle LSI

La plupart des modèles présentés dans cette section utilisent des mots-clés pour représenter le contenu d'un document (ou d'une requête). Or, il n'est pas évident que cela soit le meilleur mode de représentation. L'objectif du modèle LSI (Latent Semantic Indexing) [Deerwester *et al.*, 1990] est de transformer une représentation réalisée par mots-clé pour rapprocher les documents et requêtes sémantiquement similaires. L'index inversé résultant de l'indexation vectorielle des documents du corpus peut être "bruité", des termes pouvant n'apparaître que de manière anecdotique, ou "creux", c'est à dire contenant plutôt des mots propres à chaque document que des termes faisant la liaison entre des documents de mêmes concepts. Le modèle LSI vise alors à "nettoyer" l'index en cherchant à établir des relations entre les termes et des "concepts" sémantiques, puis entre ces concepts et les documents du corpus.

La transformation par LSI commence par appliquer un processus dit de SVD (Singular Value Decomposition) [Stewart, 1993] à une représentation vectorielle par mots-clés pour créer un nouvel espace vectoriel. Ce processus décompose l'index inversé X en trois matrices U, Σ et V :

$$X = U\Sigma V^T \tag{1.22}$$

où X est l'index inversé résultant de l'indexation vectorielle des documents du corpus, soit une matrice terme-document pondérée (par exemple par $tf * idf$, section 1.2.2) de taille $n \times Nb_D$ (nombre de termes différents dans le corpus $\times$ nombre de documents dans le corpus), U est une matrice $n \times m$ (avec $m \leq \min(n, Nb_D)$) dont chaque ligne correspond au vecteur propre $\hat{t}_i$ d'un terme, Σ est une matrice $m \times m$ dont seuls les éléments en diagonale, appelés valeurs singulières, sont non-nuls et V^T est une matrice $m \times Nb_D$ dont chaque colonne correspond au vecteur propre $\hat{D}_j$ associé au document D_j. Lorsqu'on sélectionne les k plus grandes valeurs singulières, on obtient une approximation de rang k de la matrice (ou index) originale. On réalise alors un passage des vecteurs termes $\hat{t}_i$ et documents $\hat{d}_i$ à l'espace des "concepts". Le vecteur $\hat{t}_i$ possède alors k composantes qui correspondent chacune à l'importance du terme t_i dans un concept donné. De la même manière, le vecteur $\hat{D}_i$ rend compte de l'intensité des relations entre le document D_i et chacun des k différents concepts. Cette approximation s'écrit alors sous la forme suivante :

$$X_k = U_k \Sigma_k V_k^T \tag{1.23}$$

Il est alors possible d'estimer la proximité sémantique de deux documents D_i et D_j ou de deux termes t_i et t_j en comparant leurs vecteurs respectifs ($\hat{D}_i$ et $\hat{D}_j$ pour les documents et $\hat{t}_i$ et $\hat{t}_j$ pour les termes) dans le nouvel espace vectoriel en utilisant par exemple une mesure de cosinus. Pour estimer la pertinence d'un document par rapport à une requête utilisateur, il faut traduire le vecteur de la requête dans l'espace des concepts :

$$\hat{R} = \Sigma_k^{-1} U_k^T \vec{R} \tag{1.24}$$

Le vecteur requête ainsi traduit peut alors être comparé (par exemple par une mesure de cosinus) aux vecteurs des documents dans l'espace de concepts pour identifier les documents les plus pertinents.

Ce modèle a montré des performances très intéressantes, surtout sur des corpus de petite ou moyenne taille où les résultats semblent surpasser largement ceux du modèle vectoriel classique [Dumais, 1994]. Néanmoins, sur des corpus de taille plus conséquente, les bénéfices résultant de la transformation de la matrice terme-document originale semblent s'estomper. Il existe de nombreuses variantes de ce modèle, notamment des modèles LSI probabilistes [Hofmann, 1999] qui obtiennent de très bons résultats.

Modèle de langue

Les méthodes statistiques appliquées à la linguistique informatique, notamment pour l'étiquetage syntaxique [Charniak, 1997], la reconnaissance de la parole [Jelinek, 1997], la traduction automatique [Zhao *et al.*, 2004] ou la segmentation thématique des textes [Beeferman *et al.*, 1999], connaissent un succès considérable depuis un certain nombre d'années [Manning and Schütze, 1999]. Fort de ces constatations, l'utilisation de modèles de langue en recherche d'information a été introduite par Ponte et Croft [Ponte and Croft, 1998] pour retourner à un utilisateur les documents qui correspondent le mieux à sa requête. Ces modèles cherchent à "capter" les régularités de la langue en étudiant les phrases des textes de manière statistique pour déterminer les documents les plus probablement pertinents. Bien qu'ils puissent être, à juste titre, rapprochés des modèles probabilistes, le paradigme des modèles de langue est différent : plutôt que de s'attacher à la modélisation de la notion de pertinence, on considère que la pertinence d'un document dépend de la probabilité que la requête et le document puissent être générés par un même modèle de langue, par un même sous-langage. Plus précisément, on cherche généralement à établir la probabilité que la requête ait été générée par un modèle construit en fonction du document dont on cherche à estimer la pertinence. Ainsi, le score de pertinence $Sc(D, Q)$ obtenu par chaque document D du corpus est déterminé par la probabilité $P(R|MD)$ que son modèle MD génère la requête R :

$$Sc(D, Q) = P(R|MD) \tag{1.25}$$

Un modèle de langue peut être considéré comme une fonction qui assigne une probabilité $P(s)$ à un terme ou à une séquence de termes s dans une langue donnée. En suivant de manière stricte la théorie des probabilités sur les chaînes, l'estimation de la probabilité d'apparition d'un terme dans une séquence $s = t_1^{ns} = t_1, t_2, t_3, \ldots, t_{ns}$ devrait prendre en considération l'ensemble de ses prédécesseurs. Néanmoins, afin d'estimer les probabilités de manière statistique, on ne s'intéresse généralement qu'à ses $n-1$ prédécesseurs immédiats (approximation markovienne d'ordre $n-1$). On a alors :

$$P(s) = \prod_{i=1}^{ns} P(t_i|t_{i-n+1}^{i-1}) \tag{1.26}$$

On parle alors de modèle de langue n-gramme. En particulier, les modèles unigramme ($n = 1$), bi-gramme ($n = 2$) et tri-gramme ($n = 3$) sont les plus souvent utilisés. Les probabilités $P(t_i)$ (uni-gramme), $P(t_{i-1}t_i)$ (bi-gramme) et $P(t_{i-2}t_{i-1}t_i)$ (tri-gramme) doivent alors être estimées dans la langue concernée, c'est à dire dans notre cadre de recherche d'information, dans le document ou le corpus pour lequel nous construisons le modèle. Ces probabilités peuvent être déterminées par une estimation du maximum de vraisemblance (MLE), ce qui revient, dans le cas d'un unigramme (tel que c'est le cas dans la plupart des approches en recherche d'information), à calculer la fréquence relative du terme concerné dans le document ou le corpus.

Dans l'approche proposée par Ponte et Croft [Ponte and Croft, 1998], comme dans de nombreuses autres, le score de pertinence d'un document dépend conjointement de la probabilité qu'un terme de la requête (les termes de la requête sont considérés de manière indépendante) soit généré par le modèle du document concerné et de celle qu'il soit généré par le modèle du corpus sur lequel la recherche est réalisée. L'emploi d'une telle combinaison de modèles revient à appliquer un processus de lissage (ici par interpolation), c'est à dire un assouplissement du modèle, qui vise à attribuer un score non-nul aux documents ne possédant pas l'ensemble des termes de la requête. Ce processus de lissage, qui est l'objet de nombreuses investigations [Goodman, 2001; Alvarez *et al.*, 2004], peut être vu comme un moyen d'éviter le sur-entraînement des modèles.

Les expérimentations réalisées ont montré que les modèles de langue fournissent un cadre prometteur pour appréhender la recherche d'information [Boughanem *et al.*, 2004]. De nombreux modèles de langue pour la recherche d'information existent, notamment des modèles basés sur une utilisation de chaînes de Markov cachées [Miller *et al.*, 1999], sur un calcul d'écart d'entropie (ou entropie croisée) [Lafferty and Zhai, 2001] ou sur l'emprunt de techniques propres à la traduction statistique [Berger and Lafferty, 1999]. Par ailleurs, alors que la plupart des approches utilisent des modèles de langue uni-grammes, certains travaux tentent de considérer l'interdépendance des termes de la requête en utilisant des modèles bi-grammes [Song and Croft, 1999] mais rencontrent cependant des difficultés dues au très grand nombre de paramètres à estimer. Enfin, il est à noter que ces modèles de langue ne sont pas complètement indépendants des modèles de recherche d'information traditionnels, tel que l'a fait remarquer Hiemstra en montrant que son modèle de langue pouvait se réduire au modèle vectoriel [Hiemstra, 2001].

1.3.2 Mesures de similarité

Dans cette section, nous décrivons les mesures utilisées dans les expérimentations réalisées au cours de cette thèse (voir les chapitres suivants). La mesure employée dans l'ensemble des approches que nous proposons est la mesure Cosine du populaire système SMART [Salton, 1971b]. Nous décrivons ensuite deux mesures dérivées du modèle probabiliste, la mesure Okapi [Robertson *et al.*, 1992] et la mesure Inquery [Callan *et al.*, 1995], ou plutôt des approximations de ces mesures dans le modèle vectoriel présentées dans [Singhal *et al.*, 1996b] et montrées comme ayant au moins les mêmes capacités à retourner les documents pertinents que les mesures originales. Enfin, [Singhal *et al.*, 1996b] ayant étudié l'influence de la longueur des textes sur la probabilité de pertinence des documents et sur les scores obtenus par les trois mesures de similarité des documents avec la requête, nous décrivons les observations réalisées et présentons la mesure Pivoted Cosine proposée pour améliorer les performances des systèmes.

Dans le modèle vectoriel, toute mesure de similarité peut être formulée de la

manière suivante :

$$Sim(D_i, R) = \sum_{t \in D_i \cap R} W_{D_i,t} \times W_{R,t} \tag{1.27}$$

où D_i est un document de la collection, R la requête et $W_{D_i,t}$ et $W_{R,t}$ les poids respectifs du terme t dans le document D_i et la requête R.

Selon la formulation de pondération en $tf*idf$ présentée en section 1.2.2 (formules 1.1 et 1.2) et l'expression de la mesure Cosine préconisée dans [Buckley *et al.*, 1994] et employée dans [Singhal *et al.*, 1996a], nous utilisons les pondérations suivantes pour la mesure Cosine dans l'ensemble de nos expérimentations [13] :

$$W_{D_i,t} = \frac{tf_{D_i,t}}{\sqrt{\sum_{t' \in D_i} tf^2_{D_i,t'}}} \tag{1.28}$$

$$W_{R,t} = \frac{tf_{R,t} \times idf_t}{\sqrt{\sum_{t' \in R} (tf_{R,t'} \times idf_{t'})^2}} \tag{1.29}$$

La mesure Okapi (ou mesure BM25) [Robertson *et al.*, 1992] est une mesure dérivée du modèle probabiliste par utilisation du modèle 2-Poisson pour réaliser une approximation des poids à affecter aux termes des documents et de la requête sans connaissances des ensembles de documents pertinents et non pertinents. Selon Singhal et al. [Singhal *et al.*, 1996b], cette mesure peut se décliner dans le modèle vectoriel (ce qui permet une comparaison plus facile avec la mesure Cosine) en utilisant les poids $W_{D_i,t}$ et $W_{R,t}$ suivants :

$$W_{D_i,t} = \frac{(Nb_{t \in D_i} \times \log(\frac{Nb_D - Nb_{D \ni t} + 0.5}{Nb_{D \ni t} + 0.5}))}{(2 \times (0.25 + 0.75 \times \frac{L_{D_i}}{L_{moy}}) + Nb_{t \in D_i})} \tag{1.30}$$

$$W_{R,t} = Nb_{t \in R} \tag{1.31}$$

avec $Nb_{t \in D_i}$ le nombre d'occurrences du terme t dans le document D_i, Nb_D le nombre total de documents du corpus, $Nb_{D \ni t}$ le nombre de documents du corpus contenant le terme t, L_{D_i} la taille du document D_i (en octets) et L_{moy} la taille moyenne des documents du corpus.

La mesure Inquery [Broglio *et al.*, 1994; Callan *et al.*, 1995], elle aussi dérivée du modèle probabiliste, emploie un modèle inférentiel pour estimer l'importance des termes dans les documents. Elle peut se décliner dans le modèle vectoriel en utilisant les poids $W_{D_i,t}$ et $W_{R,t}$ suivants [Singhal *et al.*, 1996b] :

13. Notons que dans cette formulation, seuls les poids des termes de la requête dépendent du facteur *idf* (pouvoir discriminant des termes).

$$W_{D_i,t} = 0.4 + \frac{0.6 \times \log\left(\frac{Nb_D}{Nb_{D\ni t}}\right)}{\log\left(Nb_D\right)} \times \left(0.4 \times H + \frac{0.6 \times \log\left(Nb_{t\in D_i} + 0.5\right)}{\log\left(\max_{t'\in D_i}(Nb_{t'\in D_i}) + 1\right)}\right) \quad (1.32)$$

$$avec\ H = 1\ si\ \max_{t'\in D_i}(Nb_{t'\in D_i}) \leq 25;\ H = \frac{25}{\max_{t'\in D_i}(Nb_{t'\in D_i})}\ sinon.$$

$$W_{R,t} = Nb_{t\in R} \quad (1.33)$$

L'approximation des mesures Okapi et Inquery dans le modèle vectoriel par ces formulations semble être performante puisque Singhal et al. [Singhal *et al.*, 1996b] rapportent, sur des corpus issus de la conférence *TREC* [TREC, 2006] (voir section 3.2), une amélioration des résultats de 2,6 % par rapport aux résultats originaux reportés dans [Robertson *et al.*, 1992] pour la mesure Okapi et de 4,7 % par rapport aux résultats originaux reportés dans [Broglio *et al.*, 1994] pour la mesure Inquery. Par ailleurs, ces approximations permettent une comparaison facilitée des mesures Okapi, Inquery et Cosine puisqu'il est alors possible de les utiliser dans un même système s'appuyant sur le modèle vectoriel. Qu'il s'agisse des comparaisons réalisées dans [Singhal *et al.*, 1996b], de celles de nombreuses campagnes d'évaluation [Robertson *et al.*, 1992; Robertson *et al.*, 1993; Robertson *et al.*, 1995] ou des expérimentations que nous reportons en section 6.2, la mesure Okapi apparaît comme la plus performante [14].

Un facteur commun à ces trois mesures est l'utilisation d'une normalisation des scores par rapport à la longueur des documents (Length Normalization Factor) afin de ne pas favoriser les documents longs qui ont plus de chances de contenir les termes de la requête (et plus de chances que chacun des termes de la requête présents dans leur texte figurent en un plus grand nombre d'occurrences). La mesure Cosine normalise les poids des termes en commun entre le document et la requête par les poids de tous les termes dans la requête et dans le document (le facteur de normalisation de longueur dans la mesure Cosine est $\sqrt{\sum_{t'\in R}(tf_{R,t'} \times idf_{t'})^2}$ pour les termes de la requête et $\sqrt{\sum_{t'\in D_i} tf^2_{D_i,t'}}$ pour les termes du document D_i), la mesure Okapi utilise un rapport entre la longueur du document concerné et la longueur moyenne des documents dans le corpus (L_{D_i}/L_{moy}) et la mesure Inquery normalise la fréquence d'occurrences des termes de la requête par la fréquence maximale d'un terme dans le document concerné ($\max_{t'\in D_i}(Nb_{t'\in D_i})$). Ces facteurs de normalisation visent à faire en sorte que les longs documents ne soient pas favorisés par rapport aux plus petits. Néanmoins, [Singhal *et al.*, 1996a] ont observé que les longs documents sont plus probablement pertinents que les petits et ont donc proposé de redonner une plus forte espérance de similarité aux longs documents à travers une mesure appelée *Pivoted Cosine*.

14. Au sens où elle produit le meilleur ordonancement des documents en fonction de la requête utilisateur (*cf.* chapitre 3 sur les mesures d'évaluation en recherche d'information).

Nous avons reproduit les expériences reportées dans [Singhal *et al.*, 1996a] afin de confirmer les observations réalisées et caractériser plus précisément le problème. Tout d'abord, il s'agit de capturer l'influence de la longueur des documents sur leur propension à être pertinents. Pour ce faire, nous utilisons les documents du corpus ZIFF avec les requêtes 1-50 de TREC-1 [TREC, 2006] pour lesquelles nous disposons de jugements de pertinence (voir section 3.2 du chapitre suivant). Ainsi que le propose [Singhal *et al.*, 1996a], nous ordonnons les 75000 documents les plus courts du corpus (sur 75180) selon leur nombre de termes significatifs puis nous divisons la liste ainsi obtenue en 75 groupes de documents (les documents d'un même groupe sont alors de tailles relativement proches). La probabilité qu'un document d'une taille donnée soit pertinent peut alors être estimée statistiquement en comptant, pour l'ensemble des requêtes utilisées (44 requêtes ayant chacune au moins un document pertinent dans le corpus), le nombre de documents pertinents présents dans le groupe correspondant et en divisant cette somme par le nombre total de documents pertinents (2703 documents). Parallèlement, nous estimons l'influence de la longueur des textes sur les mesures d'estimation de pertinence afin de pouvoir la comparer à celle qu'elle a sur les probabilités de pertinence effective. À l'instar de [Singhal *et al.*, 1996a], nous cherchons à évaluer la probabilité qu'a un document d'une taille donnée d'être retourné dans les 1000 premiers documents de la liste de résultats (appelée par la suite probabilité de sélection). Pour ce faire, nous procédons de la même manière que pour l'estimation de la probabilité de pertinence effective, en formant des groupes de documents selon leur taille et en comptant le nombre de documents de chaque groupe à être retournés dans les 1000 premiers pour chacune des 44 requêtes utilisées. Le total obtenu pour chaque groupe est finalement divisé par le nombre total de documents retournés, c'est à dire 44000.

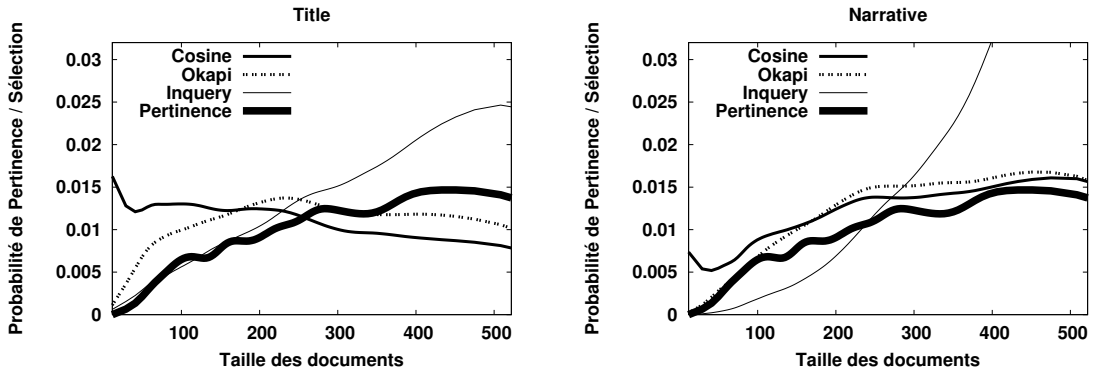

FIGURE 1.5 – Probabilité de pertinence / sélection en fonction de la taille des documents

La figure 1.5 trace, en fonction de la taille des documents (en réalité la taille moyenne des groupes de documents), les distributions de probabilités obtenues en

utilisant des requêtes courtes (Title) formées de quelques mots-clés (graphique de gauche) et des requêtes plus longues (Narrative) comprenant la description narrative de ce que l'on recherche (graphique de droite) [15]. L'examen des courbes obtenues confirme les observations réalisées dans [Singhal *et al.*, 1996b], la probabilité de pertinence des documents augmente avec leur taille. Dans cet article, les auteurs

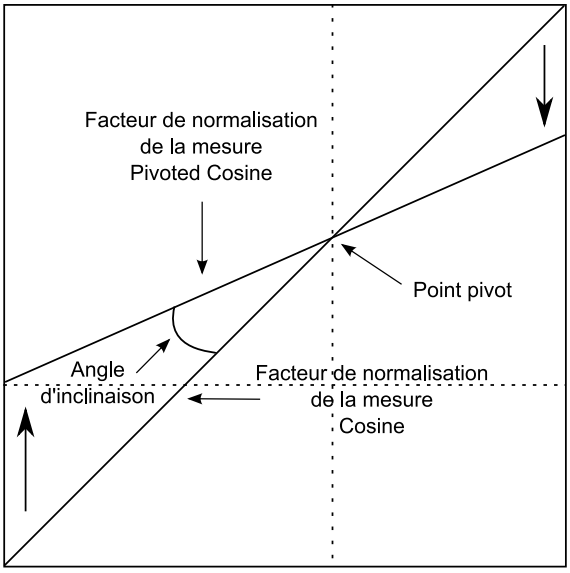

FIGURE 1.6 – Inclinaison du facteur de normalisation de la mesure Cosine.

imputent les meilleures performances de la mesure Okapi au fait que la probabilité de sélection des documents suit mieux les variations de probabilité de pertinence avec cette mesure qu'avec les autres. Avec la mesure Cosine, lorsque des requêtes courtes sont utilisées, la probabilité de sélection des petits documents est en effet bien supérieure à leur probabilité de pertinence et inversement, la probabilité de sélection des longs documents est bien inférieure à leur probabilité de pertinence. Singhal et al. ont alors proposé de modifier cette mesure pour faire coïncider les deux probabilités. L'idée est de réduire l'impact du facteur de normalisation pour les documents contenant un grand nombre de termes et de l'augmenter pour ceux qui en contiennent peu. L'objectif est alors de faire pivoter le facteur de normalisation des poids des termes des documents (figure 1.6). Un point de pivot p et un degré d'inclinaison α doivent alors être choisis pour calculer les nouveaux poids des termes :

$$W_{D_i,t} = \frac{tf_{D_i,t}}{(1-\alpha) \times p + \alpha \times \sqrt{\sum_{t' \in D_i} tf_{D_i,t'}^2}} \tag{1.34}$$

15. Les mots Title et Narrative correspondent aux noms des champs des fichiers de requêtes de TREC, voir section 3.2.

Le point pivot préconisé dans [Singhal *et al.*, 1996a] correspond au facteur de normalisation moyen calculé en considérant l'ensemble des documents du corpus. Le degré d'inclinaison, quant à lui, a été fixé empiriquement à la valeur 0.2, valeur qui semble permettre l'obtention des meilleurs résultats dans [Singhal *et al.*, 1996a]. Néanmoins, Chung et al. [Chung *et al.*, 2006] font remarquer que, tel qu'observé en comparant les deux graphiques de la figure 1.5, la taille de la requête a elle aussi un impact sur la probabilité de sélection des documents. La taille de la requête QL (selon le nombre de termes de la requête différents) doit alors être prise en compte dans le calcul du degré d'inclinaison. Chung et al. ont donc déterminé une équation pour estimer le degré d'inclinaison optimal :

$$\alpha = 0.0921 \times \log(QL) + 0.0658 \tag{1.35}$$

Cette estimation permettant d'améliorer significativement les résultats lorsque la requête dépasse une dizaine de termes, nous utilisons ce calcul du degré d'inclinaison optimal dans nos expérimentations présentées dans les chapitres suivants.

La normalisation des mesures de similarité selon la taille des textes comparés est un problème complexe qui a motivé de nombreux travaux. Bien qu'elle semble permettre d'améliorer significativement les performances des systèmes de recherche classiques, nous ne sommes pas pleinement convaincus des effets bénéfiques de la mesure Pivoted Cosine, notamment lorsqu'il s'agit de l'appliquer dans le cadre des approches s'intéressant aux passages des documents (approches de Passage Retrieval [Kaszkiel and Zobel, 2001]) . Nous discuterons plus avant de cette mesure au chapitre 6.

1.4 Formulation d'une recherche d'information

La formulation d'une recherche est un processus cognitif complexe qui fait appel à de nombreuses connaissances, non seulement sur le sujet pour lequel on a un besoin d'information et sur la base documentaire que l'on interroge, mais aussi sur le système que l'on utilise. Il faut en effet savoir comment exprimer son besoin d'information de manière claire et compréhensible par le système de recherche d'information. La formulation d'un besoin est le point de départ de toute recherche d'information. C'est en effet elle qui définit l'objectif à atteindre. Généralement, cela se fait par le biais d'une requête (ou question). Selon Kleinberg [Kleinberg, 1999], il existe trois différentes formes de requêtes :

- Les requêtes spécifiques : *e.g.*, "Qui a obtenu le prix Nobel de la paix en 2002 ?" ;
- Les requêtes larges : *e.g.*, "Trouver des informations sur les prix Nobel de la paix" ;
- Les requêtes par similarité : *e.g.*, "Trouver des documents en relation avec la page Wikipédia sur les prix Nobel de la paix".

Bien que ces trois formes de requêtes impliquent des traitements différents, une même question se pose dans tous les cas : quel langage utiliser pour que le besoin soit interprété de manière satisfaisante ? Le mode de formulation d'une requête, qui est un problème soulevant de nombreuses interrogations, est exploré dans la première partie de cette section. Ensuite, puisqu'il n'est pas évident que l'utilisateur soit toujours capable d'exprimer son besoin de manière précise lors de la formulation de sa requête, nous présentons brièvement quelques techniques utilisées par certains systèmes pour améliorer la définition des informations recherchées.

1.4.1 Types de requêtes

La plupart des systèmes de recherche d'information (par exemple Google [Google, 2008]) proposent à l'utilisateur de formuler son besoin sous la forme d'une liste de mots-clés, avec ou sans spécification d'opérateurs booléens. Ces opérateurs booléens offrent une grande flexibilité aux usagers, leur permettant de spécifier la plupart des besoins de manière précise. Néanmoins, il s'avère que la plupart des utilisateurs de systèmes de recherche d'information rencontrent des difficultés à utiliser correctement les connecteurs logiques "et" et "ou" [Dinet, 2000]. Par exemple, un utilisateur s'intéressant au modèle probabiliste et au modèle vectoriel peut, en réalité, être intéressé par les documents traitant du modèle probabiliste *ou* du modèle vectoriel. Dans un tel cas, la plupart des utilisateurs ont tendance à utiliser l'opérateur logique "et" à la place du "ou" qu'ils auraient dû employer pour obtenir l'ensemble des documents qui les intéressent. Par ailleurs, ce langage, qui est parfaitement adapté au traitement informatique de données, n'offre pas l'étendue des possibilités d'expression du langage naturel.

De nombreuses approches ont été proposées pour permettre de spécifier plus précisément un besoin d'information. Par exemple, de nombreux travaux ont tenté d'étendre le langage booléen pour être capable d'interpréter des requêtes plus complexes. Pour la recherche sur Internet, le plus répandu de ces langages est certainement le langage WebSQL, introduit par Mihaila dans [Mihaila, 1996], qui se base sur un formalisme similaire à celui du langage SQL d'interrogation de bases de données. Ce langage, qui permet une spécification précise de ce que l'on recherche, n'est cependant pas facilement abordable lorsque l'on ne connait pas le langage SQL ni le fonctionnement d'une base de données relationnelle. Tout langage différant du langage naturel nécessite un apprentissage de la part de l'utilisateur.

Une autre piste consiste alors à analyser une question posée directement en langage naturel. De nombreuses approches tentent d'utiliser des techniques propres au traitement de la langue naturelle pour interpréter la requête formulée et la transposer sous une forme interprétable par le modèle de recherche d'information utilisé [16]

16. Notons que le modèle vectoriel accepte des requêtes posées en langue naturelle mais ne les interprète pas toujours correctement. Dans la forme classique du modèle, il n'y a en effet pas d'interprétation préliminaire de la langue, la requête est représentée de la même manière que les documents, par extraction des termes informatifs. Par exemple, la négation n'est pas prise en considération.

[Lallich-Boivin *et al.*, 2006]. Un exemple de système réalisant une interprétation des requêtes formulées en langage naturel est le système Mulder proposé par Kwok dans [Kwok *et al.*, 2001] qui construit un arbre sémantique en fonction de la structure de la phrase puis utilise un classifieur et des règles spécifiques à la langue pour produire un certain nombre de requêtes sous forme de mots-clés, allant d'une forte spécialisation (utilisant beaucoup de termes de la question initiale), à une requête générale (contenant les termes les plus importants de la question). Mulder a montré de bonnes performances mais ses résultats restent néanmoins inférieurs à ceux obtenus lorsqu'une requête booléenne est formulée directement par l'utilisateur.

Malgré les efforts réalisés pour permettre à l'utilisateur non-expert d'exprimer ses besoins de manière précise, la recherche par requêtes booléennes reste à l'heure actuelle l'approche la plus efficace. Néanmoins, l'expansion automatique de requêtes (section 1.4.2) tente de combler les lacunes de la requête initiale en reformulant la question posée selon les documents contenus par la base documentaire et les appréciations de pertinence réalisées par l'utilisateur. Par ailleurs, tel que détaillé en section 1.5, la présentation des résultats joue un rôle primordial dans la recherche : un regroupement des documents par classes thématiques, ou une présentation graphique des résultats, peut permettre à l'utilisateur d'orienter efficacement sa recherche vers les informations qui l'intéressent malgré une requête initiale peu précise.

1.4.2 Expansion de requêtes

L'expansion automatique (ou reformulation) de requêtes vise à transformer la requête initiale formulée par l'utilisateur pour permettre un meilleur ciblage de ses besoins et donc de réaliser une meilleure collecte des informations recherchées. Deux grandes familles d'approches de reformulation de requêtes existent [Manning *et al.*, 2008] :

- Les approches globales qui utilisent des ressources indépendantes de la requête pour la reformuler.
- Les approches locales qui reformulent la requête selon les documents qui semblent correspondre aux besoins de l'utilisateur (en utilisant des appréciations de pertinence).

Approches globales

La plupart des systèmes de recherche d'information travaillent par utilisation de mots-clés. Ces mots-clés sont supposés représenter un concept (ou une partie de concept) abordé par un document ou une requête. Néanmoins, en raison du au polymorphisme de la langue naturelle, ainsi que de l'ambiguïté de certains concepts, un mot-clé n'est pas toujours l'unique représentant d'un concept et un concept n'est pas toujours la signification unique d'un mot-clé. Afin de prendre en compte cette observation, il est possible de réaliser un traitement sur la requête en ajoutant/supprimant des termes selon des ressources linguistiques externes ou des statistiques sur le fonds

documentaire interrogé. Deux questions majeures se posent alors : Quels termes doit-on utiliser pour étendre la requête ? Comment les nouveaux termes doivent-ils être ajoutés dans la requête (pondération) ?

De nombreuses méthodes d'expansion globale de requête existent. Parmi elles, notons l'approche proposée par Voorhees [Voorhees, 1994], qui utilise un thésaurus (le thésaurus WordNet [Miller, 1995]) pour ajouter les synonymes des termes de la requête initiale, l'approche proposée par Qiu et Frei [Qiu and Frei, 1993], qui étudient le corpus pour établir des relations de synonymie selon les co-occurrences de termes (construction automatique de thésaurus), ou l'approche proposée par Cucerzan et Brill [Cucerzan and Brill, 2004], qui tente de corriger les termes mal orthographiés de la requête selon des statistiques réalisées sur le corpus de textes utilisé.

Le fait d'ajouter des termes à une requête conduit généralement à l'obtention d'un plus grand nombre de résultats (si l'on considère comme résultat, tous les documents contenant au moins un terme en commun avec la requête). L'ensemble de résultats proposé à l'utilisateur a alors plus de chances de contenir des documents pertinents. D'un autre côté, le nombre de résultats étant plus important, il sera plus difficile de les localiser si ceux-ci n'apparaissent pas en tête de liste. Une grande attention doit alors être portée sur les termes à ajouter. Certaines approches proposent une liste de termes potentiellement en relation avec les besoins d'information et attendent une intervention de l'utilisateur pour les ajouter à la requête [Manning *et al.*, 2008].

Approches locales

De nombreux travaux ont cherché à modéliser les processus mis en jeu lors d'une recherche d'information afin de produire des systèmes permettant d'aider efficacement l'utilisateur à trouver ce qui l'intéresse. Deux grands courants se distinguent : la recherche d'information peut être perçue comme une résolution de problème - on cherche à atteindre un but clairement défini [Marchionini, 1995] - ou comme une activité exploratoire, c'est à dire que la recherche s'oriente selon les informations que l'on récolte [Bates, 1989]. Les approches locales d'expansion automatique de requêtes (ou reformulation de requêtes) [Ruthven *et al.*, 2001; Ruthven and Lalmas, 2003; Bai *et al.*, 2005] se situent plutôt autour de la recherche d'information vue comme une activité exploratoire puisque l'objectif est de parer aux lacunes de la requête initiale en utilisant les documents du corpus ainsi que des interactions avec l'utilisateur pour retourner les informations qui correspondent le mieux au besoin établi. Belkin [Belkin *et al.*, 1997], se basant sur les idées et conceptions formulées dans [Paisley and Parker, 1965] et [Taylor, 1968] , souligne le paradoxe qui affecte toute recherche d'information (phénomène *ASK*, "*Anomalous State of Knowledge*") : pour qu'un système puisse répondre de manière idéale à un besoin d'information, la requête de l'utilisateur doit comporter toutes les informations que l'utilisateur recherche. Or, si l'utilisateur recherche des informations, c'est justement qu'il n'en connaît pas l'entière étendue *a priori*. Il ne peut donc pas exprimer son besoin de manière

précise et risque alors de ne pas obtenir une réponse idéale. Cette constatation renforce l'idée qu'une communication entre l'utilisateur et le système peut avoir des effets bénéfiques sur l'efficacité de la recherche d'information. Ainsi, la plupart des approches locales d'expansion de requête utilisent des retours de pertinence, c'est à dire qu'elles prennent en considération des appréciations de pertinence réalisées par l'utilisateur sur des documents préalablement retournés par le système pour recadrer la recherche.

Les systèmes utilisant une réinjection de pertinence suivent typiquement le processus suivant [17] :

1. Le système produit une liste ordonnée de documents en fonction de la requête initiale formulée par l'utilisateur.
2. L'utilisateur examine les k premiers documents de la liste fournie et détermine quels en sont les documents pertinents et non pertinents.
3. Le système utilise les informations recueillies pour modifier la requête initiale, en augmentant le poids (ou en ajoutant) des termes appartenant aux documents pertinents et en diminuant le poids (ou en supprimant) des termes qui appartiennent aux documents non pertinents.
4. Le système produit alors une nouvelle liste ordonnée de documents en fonction des nouveaux termes ou nouveaux poids des termes. Le processus peut être réitéré tant que l'utilisateur n'est pas pleinement satisfait des informations recueillies.

De très nombreuses approches de reformulation de requête utilisant des *feedbacks* utilisateurs (retours de pertinence) existent. La première formalisation de la technique de réinjection de pertinence dans le modèle vectoriel est celle de Rocchio [Rocchio, 1971]. Sa définition du problème, en tant que recherche de la requête optimale, l'a conduit à proposer une formule pour étendre la requête de l'utilisateur selon les jugements de pertinence :

$$\vec{R'} = \vec{R} + \frac{1}{n_p}\sum_{i=1}^{n_p}\vec{P_i} - \frac{1}{n_{np}}\sum_{i=1}^{n_{np}}\vec{NP_i} \tag{1.36}$$

où R et R' correspondent respectivement à l'ancienne et à la nouvelle requête, $\vec{P_i}$ au vecteur du i-ème document pertinent identifié, $\vec{NP_i}$ au vecteur du i-ème document non pertinent identifié et n_p et n_{np} au nombres de documents pertinents et non pertinents examinés. Cette approche, qui a montré une bonne capacité à retrouver les documents pertinents non encore examinés, a depuis inspiré de nombreux travaux. Par ailleurs, il est à noter que de nombreux modèles, notamment les modèles probabilistes décrits en section 1.3.1, sont dédiés à la réinjection de pertinence puisqu'ils

17. Excepté les approches réalisant un réinjection dite de pseudo-pertinence [Croft and Harper, 1997], qui ne font pas intervenir l'utilisateur mais supposent que les premiers documents retournés par la recherche initiale sont pertinents, et les approches à réinjection de pertinence implicite [Jung *et al.*, 2007], qui tirent leurs informations de pertinence des parcours des utilisateurs dans les réseaux.

s'appuient sur un des ensembles de documents jugés pertinents et non pertinents pour une requête afin d'estimer leurs probabilités de pertinence.

1.5 Présentation des résultats

Le choix d'un mode de présentation des résultats est un point primordial lors de la conception d'un système de recherche d'information. En effet, le moteur de la recherche d'information a beau être très performant, le système ne sera pas utilisable s'il ne permet pas d'appréhender facilement les résultats qu'il présente. Étant donnée une requête, un ensemble de documents et des estimations de similarité entre les différentes entités, la question est de trouver la meilleure façon de structurer les informations pour permettre leur interprétation par l'utilisateur du système. Selon Mann [Mann, 1999], le choix d'une structure particulière à imposer aux résultats d'une recherche dépend de quatre facteurs (environnement des 4 T) : Target user group (le groupe d'utilisateurs ciblé), Type and number of data (le nombre et la nature des documents de la base interrogée), Task to be done (la tâche à accomplir, ce pour quoi le système est utilisé) et Technical possibilities (les possibilités techniques à disposition). Tel que l'a fait remarquer Washburne [Washburne, 1927], il n'existe pas de présentation efficace pour toutes les données et tous les utilisateurs. Néanmoins, selon la destination finale du systèmes, il existe des modes de présentation plus adéquats que d'autres. De nombreuses méthodes d'affichage des résultats, de la plus simple (liste ordonnée de documents) à la plus sophistiquée (par exemple les visualisations graphiques "cone-trees" et "tree-maps" présentées dans [Hearst and Karadi, 1997] et [Shneiderman, 1992]), ont été proposées dans la littérature, chacune possédant ses propres caractéristiques qui la rendent plus ou moins adaptée à un système donné. Cette section présente maintenant rapidement les différents types de structuration et de visualisation utilisés dans les systèmes de recherche d'information actuels.

Traditionnellement, un système de recherche d'information présente ses résultats sous la forme d'une liste ordonnée de documents, ou plutôt de titres de documents généralement accompagnés de quelques lignes contenant les termes de la requête (voir par exemple le moteur de recherche Google [Google, 2008]). Les documents de la liste sont triés par ordre de score de pertinence attribué par la mesure de similarité utilisée (section 1.3) et l'utilisateur doit parcourir linéairement la liste jusqu'à avoir le sentiment d'avoir récolté suffisamment d'information pour satisfaire ses besoins. Ce type de présentation est employé par la plupart des systèmes de recherche d'information [Voorhees and Harman, 1997]. Néanmoins, tel que mentionné en introduction de ce rapport, il est difficile pour l'utilisateur de savoir quand arrêter sa collecte d'information. Comment être sûr que l'information que l'on recherche ne se trouve pas dans les prochains documents non encore examinés ? L'utilisateur doit prendre la décision d'arrêter sa collecte d'information alors qu'il ne connait pas la diversité des textes en relation avec sa requête. Par ailleurs, un tel parcours de liste

peut s'avérer très fastidieux. Des études statistiques ont montré que généralement les utilisateurs de systèmes employant ce type de présentation des résultats ne s'intéressent qu'aux tout premiers documents de la liste [Silverstein *et al.*, 1998; Spink *et al.*, 2001] [18], considérant alors que si les informations qu'ils recherchent ne se trouvent pas dans les premiers documents examinés, ils ont peu de chances de les trouver dans les documents suivants. Ce qui n'est bien sûr pas toujours vrai étant donné que les estimations de pertinence réalisées par les systèmes ne sont pas infaillibles et que, pour diverses raisons, les documents contenant l'information recherchée peuvent être précédés dans la liste par des documents déconnectés du besoin de l'utilisateur.

Une présentation des résultats par liste ordonnée de documents n'est alors peut être pas le meilleur moyen d'aider l'utilisateur dans sa recherche. Toutefois, notons que l'ajout d'informations dans la liste présentée peut permettre à l'utilisateur de cerner plus facilement le contenu des documents qui lui sont retournés [Drori, 2000]. Par exemple, afin de permettre une meilleure interprétation des résultats, il est possible d'afficher des barres de visualisation pour caractériser les critères qui justifient le classement de chaque document dans la liste [Hearst, 1994; Hearst, 1995]. La figure 1.7 donne deux exemples de barres de visualisation du contenu couramment affichées par les moteurs de recherche actuels, sous forme de courbes (a) ou sous forme de damier (b). Ces deux types de barres de visualisation

FIGURE 1.7 – Deux exemples de barres de visualisation du contenu

prennent une taille qui dépend de la longueur du document concerné et représente les positions des mots-clés recherchés. Avec la visualisation par courbes, chaque mot-clé recherché est associé à une couleur particulière (ici deux mots-clés) et la hauteur de la courbe correspondante renseigne sur la densité de présence de ce mot-clé dans les différentes parties du document. Dans la barre de visualisation sous forme de damier, chaque ligne correspond à un mot-clé et l'intensité de la couleur de chaque point d'une ligne donnée est déterminée par la fréquence du mots-clé correspondant dans la zone de texte concernée (une couleur sombre correspond à un fréquence d'occurrences importante). L'ajout de telles informations à la liste ordonnée de résultats permet d'appréhender rapidement l'organisation générale de chaque document retourné par le système et augmente significativement la capacité de l'utilisateur à

18. Ces études ont montré que qu'une écrasante majorité des utilisateurs n'examinent pas plus d'une dizaine de descriptions de documents.

identifier les documents qui l'intéressent [Drori, 2000].

Néanmoins, il semble que les apports les plus importants concernent l'organisation des résultats et la mise en évidence des relations existant entre les documents retournés. Les études menées dans [Drori and Alon, 2003] et [Chen and Dumais, 2000] montrent que les systèmes regroupant les documents d'une liste de résultats par catégories apportent un réel avantage aux utilisateurs. Regrouper les documents fortement similaires permet de réduire de moitié le temps passé à rechercher l'information désirée lorsque celle-ci est contenue dans la liste de résultats. De plus, il ressort de ces études que les utilisateurs ont plus facilement confiance dans la fiabilité des informations qu'ils ont pu extraire lorsque celles-ci sont contenues par des documents appartenant à une catégorie thématique.

Les relations existant entre les documents peuvent être mises en évidence de diverses manières. Dans les études menées dans [Drori and Alon, 2003] et [Chen and Dumais, 2000], les documents similaires sont simplement regroupés dans la liste de résultats mais ce n'est pas la seule façon possible de rendre compte des ressemblances thématiques. Dans un cadre de présentation purement textuelle, de nombreuses approches ont été proposées pour orienter l'utilisateur dans sa recherche d'information. Chimera distingue trois types de visualisations textuelles de catégories et sous catégories [Chimera and Shneiderman, 1994] :

- Présentation d'une liste complète de catégories et sous-catégories dans laquelle l'utilisateur navigue grâce à une barre de défilement,
- Présentation par liste dynamique, où il est possible de développer et réduire les catégories (de manière similaire à un explorateur de fichier Windows),
- Présentation par panneaux multiples où chaque panneau est dédié à un niveau de profondeur de la catégorisation.

L'étude réalisée dans [Chimera and Shneiderman, 1994] a montré que le fait de présenter la totalité de la hiérarchie à l'utilisateur pouvait s'avérer très perturbant. Il semble préférable d'utiliser une présentation par panneaux multiples.

Zamir et Etzioni définissent trois facteurs déterminants de la qualité d'un système de recherche d'information réalisant une catégorisation (ou clustering) de ses résultats [Zamir and Etzioni, 1998] :

- Les catégories (recouvrantes ou non) doivent être des groupes de résultats cohérents : une attention particulière doit être portée sur la méthode de clustering choisie ainsi que sur les modes de calcul de similarité entre documents,
- Le système doit présenter des descriptions précises des catégories produites : l'objectif est de permettre à l'utilisateur de naviguer facilement dans les groupes en lui donnant les moyens de déterminer rapidement les catégories susceptibles de satisfaire ses besoins,
- La catégorisation des résultats ne doit pas significativement ralentir le système : il faut trouver un compromis entre qualité des groupes et rapidité d'affichage.

Un exemple de système réalisant une présentation des catégories à l'utilisateur est le système Grouper [Zamir and Etzioni, 1999] qui allie description du contenu

des catégories thématiques par extraction de mots-clés et sélection de documents représentatifs des différents groupes (voir section 2.4 pour la représentation du contenu des groupes). Cette présentation mixte permet à l'utilisateur de cerner rapidement de quoi chaque catégorie retourne sans surcharger la page de résultats. Un autre système célèbre est le système "Scatter/Gather" présenté dans [Cutting *et al.*, 1992] qui propose une interaction avec l'utilisateur pour obtenir un groupe final de documents idéal. Initialement, il présente une courte description des catégories de documents automatiquement produites et propose à l'utilisateur de sélectionner les groupes qui l'intéressent. Les groupes sélectionnés sont alors fusionnés pour appliquer aux sous-ensembles de documents ainsi obtenus un nouveau processus de clustering permettant de présenter une nouvelle liste de descriptions de groupes thématiques. L'utilisateur peut réitérer le processus tant qu'il n'a pas obtenu un groupe de documents satisfaisant pleinement ses besoins. Le système a montré une bonne capacité à orienter les utilisateurs vers les informations qui les intéressent, surtout lorsque leurs besoins concernent des informations bien spécifiques [Pirolli *et al.*, 1996]. Ce système peut se comparer aux techniques de réinjection de pertinence (voir section 1.4.2).

Alternativement à la présentation purement textuelle des résultats, de nombreux travaux ont tenté de cerner les éléments visuels pouvant permettre d'exploiter pleinement les résultats d'une recherche d'information. Ces systèmes présentent généralement, dans des espaces à deux ou trois dimensions, des informations concernant les relations existant entre les documents retournés par le système de recherche d'information. Ces informations peuvent concerner l'appartenance des documents à des catégories données ou simplement représenter leurs distances thématiques. Le lecteur intéressé pourra se référer à [Picarougne, 2004] qui réalise un bon aperçu des différentes approches de visualisation graphique existantes.

De nombreuses études se sont intéressées aux bénéfices résultant de l'utilisation d'approches de visualisations graphiques des résultats en deux [Nowell *et al.*, 1996; Veerasamy and Belkin, 1996; Veerasamy and Heikes, 1997] et en trois dimensions [Lamping *et al.*, 1995; Swan and Allan, 1998; Sebrechts *et al.*, 1999], et plus particulièrement au rapport entre complexité d'affichage et niveau d'expertise des utilisateurs. L'ensemble de ces études concluent que tout mode de présentation des résultats nécessite un apprentissage de la part de l'utilisateur et que plus le système réalise un affichage graphiquement évolué, contenant un important volume d'informations, plus son adoption par le public risque d'être difficile. Lors de la conception d'un outil d'aide à la recherche d'information, il faut alors bien garder à l'esprit que la simplification des informations à transmettre est un point clé de l'utilisabilité du futur système. Quel que soit le mode de présentation utilisé, les informations affichées ne doivent pas être trop abondantes pour permettre à l'utilisateur de les interpréter. Un trop gros volume d'informations peut nuire à la compréhension des résultats, surtout lorsque l'utilisateur n'est pas habitué au système utilisé [Cugini *et al.*, 2000].

1.6 Conclusion

La recherche d'information est une branche de recherche très active qui a fait l'objet d'une multitude de travaux depuis le début du siècle dernier. Face à la masse de données disponibles sur les réseaux et dans les bases de données, il devient en effet aujourd'hui indispensable de disposer d'outils performants pour localiser l'information répondant à nos besoins. Dans ce chapitre, nous avons présenté les concepts de base de la recherche d'information traditionnelle, en passant en revue les fondements des principaux systèmes de recherche d'information actuels. Les différentes composantes d'un système de recherche d'information ont alors été détaillées, de l'expression d'un besoin à la présentation des résultats en passant par les modes de représentation des documents et les mesures d'estimation de pertinence.

L'ensemble des travaux reportés dans la suite de cette thèse s'appuient alors sur des concepts énoncés dans ce chapitre : nous nous plaçons dans un contexte de recherche d'information statique [19] basée sur le modèle vectoriel (section 1.3.1), où les unités d'indexation (mots simples) utilisées pour représenter les textes sont obtenues par application du processus de lemmatisation détaillé dans [Porter, 1980] (section 1.2.1) et élimination des mots vides figurant dans la stop-liste proposée par [Fox, 1992] (section 1.2.2).

En dernière section de ce chapitre, nous avons vu que le choix d'un mode de présentation des résultats à un utilisateur est un problème complexe qui requiert une grande attention lors de l'élaboration d'un système de recherche d'information. Dans la plupart des études réalisées, il a été observé que, d'une manière générale, le regroupement des résultats en catégories thématiques est un mode de présentation relativement bien accueilli par les utilisateurs de systèmes de recherche d'information, quel que soit leur niveau d'expertise. La catégorisation des résultats présente en effet l'avantage d'être relativement intuitive, le regroupement d'objets selon leurs similitudes étant un concept que l'on appréhende dès le plus jeune âge. Dans le chapitre suivant, nous nous intéressons alors aux différents concepts et techniques mis en jeu dans les processus de clustering de documents appliqués à la recherche d'information.

19. C'est à dire sans interaction avec l'utilisateur ni expansion de requêtes. Il est à noter néanmoins que, tel qu'énoncé en conclusion de ce rapport, il paraît tout à fait envisageable d'étendre les observations réalisées et approches présentées pour les inclure dans un contexte plus interactif.

Chapitre 2

Clustering et recherche d'information

Offrant des alternatives plus qu'intéressantes à la classique liste ordonnée de documents, les techniques de catégorisation automatique permettent bien souvent de réduire les efforts à fournir pour localiser l'information recherchée. Ce chapitre explore les concepts sur lesquels de tels processus se basent pour aider les utilisateurs dans leurs recherches. Nous commençons par réaliser une présentation générale des techniques de clustering puis nous discutons de leur mise en application dans le domaine de la recherche d'information, en s'interrogeant sur les mesures de distances à considérer, les divers modes de fonctionnement des méthodes et les différentes manières de décrire les catégories produites.

Sommaire

2.1 Présentation et applications 48
2.2 Relations entre documents 52
2.3 Méthodes de clustering 53
2.3.1 Mode de représentation des clusters 55
2.3.2 Méthodes non-hériachiques 57
2.3.3 Méthodes hiérarchiques 60
2.4 Description du contenu des groupes thématiques 63
2.5 Conclusion 66

2.1 Présentation et applications

Un algorithme de clustering (ou catégorisation) cherche à faire émerger la structure d'un ensemble de données en déterminant des groupes d'éléments proches selon des relations de distances pré-établies [Gordon, 1987; Willett, 1988]. Le but est de partitionner un ensemble de données $\mathcal{D} = \{D_1, \ldots, D_n\}$ en k sous-ensembles (ou clusters) $\mathcal{C} = \{C_1, \ldots, C_k\}$ tels que [1] :

$$\begin{cases} \forall i \in \{1, \ldots, k\}, C_i = \{C_i^1, \ldots, C_i^{|C_i|}\} \neq \emptyset, \\ \forall i, j \in \{1, \ldots, k\}^2, i \neq j \Rightarrow C_i \cap C_j = \emptyset \\ \bigcup_{i=1}^{k} C_i = \mathcal{D} \end{cases} \tag{2.1}$$

Si l'on dispose de mesures de proximité $Sim(D_i, D_j) \in [0, 1]$ entre deux données D_i et D_j, un algorithme de clustering cherche généralement à maximiser la cohésion (c'est à dire la similarité moyenne entre éléments de même groupe) et la séparation des clusters (c'est à dire la dissimilarité moyenne entre groupes différents). Par exemple, avec $Cl(D_i, D_j)$ une fonction qui retourne 1 si les données D_i et D_j appartiennent au même cluster (et 0 sinon), les calculs de cohésion et de séparation peuvent être les suivants :

$$Cohesion = \frac{\sum_{D_i \in \mathcal{D}} \sum_{\substack{D_j \in \mathcal{D}, \\ i \neq j}} Cl(D_i, D_j) \times Sim(D_i, D_j)}{\sum_{D_i \in \mathcal{D}} \sum_{\substack{D_j \in \mathcal{D}, \\ i \neq j}} Cl(D_i, D_j)} \tag{2.2}$$

$$Separation = \frac{\sum_{D_i \in \mathcal{D}} \sum_{\substack{D_j \in \mathcal{D}, \\ i \neq j}} (1 - Cl(D_i, D_j)) \times (1 - Sim(D_i, D_j))}{\sum_{D_i \in \mathcal{D}} \sum_{\substack{D_j \in \mathcal{D}, \\ i \neq j}} (1 - Cl(D_i, D_j))} \tag{2.3}$$

Ces formulations de cohésion et de séparation des clusters ne sont que des exemples. Tel que nous le verrons en section 2.3, une multitude de méthodes de clustering existent, chacune possédant ses propres objectifs ou critères à maximiser. La figure 2.1 illustre le partitionnement d'un ensemble de données dans un espace à deux dimensions. Ici, trois groupes se dégagent clairement puisqu'on distingue nettement trois zones de l'espace contenant des éléments relativement proches. Malheureusement, une telle configuration n'est que rarement observée dans la réalité, la distinction entre groupes de données s'avère parfois bien plus difficile. Parmi les méthodes permettant la production de tels groupes, on distingue les méthodes supervisées (ou méthodes de classification), qui nécessitent un apprentissage à partir

1. Cette définition n'est valable que pour les clustering de type *Hard*, dont les groupes produits ne se recouvrent pas. La définition d'un clustering de type *Soft* ne contient pas la deuxième condition du système 2.1.

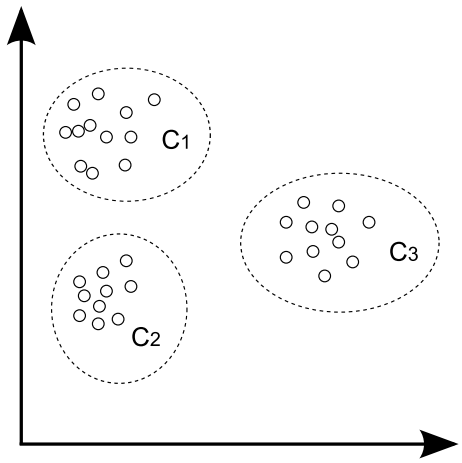

FIGURE 2.1 – Clustering de données dans un espace en deux dimensions

de données pré-classées (par exemple, par utilisation de réseaux neuronaux ou des célèbres *Support Machine Vectors* [Platt, 1999]), des méthodes non-supervisées, qui tentent de déterminer des groupes de données sans apport de connaissances et sans définition préliminaire des groupes à construire [Manning *et al.*, 2008]. Bien que les méthodes supervisées puissent s'avérer très utiles en recherche d'information, notamment pour classer des documents dans des classes pré-définies (voir par exemple [Dumais *et al.*, 1998]), nous nous intéresserons uniquement aux méthodes de clustering non-supervisées dans la suite de ce chapitre (et de ce rapport), notre objectif étant de caractériser automatiquement les différents aspects du sujet de l'utilisateur à partir de sa requête et des informations contenues dans le corpus de textes.

Les techniques de clustering non-supervisées trouvent de très nombreuses applications en recherche d'information. La plupart d'entre elles s'appuient en grande partie sur la *Cluster Hypothesis* introduite par Jardine et Van Rijsbergen dans [Jardine and Van Rijsbergen, 1971], qui stipule que les documents thématiquement proches tendent à être pertinents aux mêmes requêtes. Il en découle que les documents pertinents à une requête donnée tendent à être plus proches les uns avec les autres qu'avec les documents non pertinents et que donc, ils ont de grandes chances de figurer dans le même cluster [Van Rijsbergen, 1979]. Manning et al. [Manning *et al.*, 2008] distinguent quatre applications majeures des techniques de clustering en recherche d'information qui se fondent en grande partie sur cette hypothèse. Nous reportons ces applications dans la table 2.1.

Traditionnellement, les méthodes de clustering étaient appliquées de manière statique, c.a.d. sur le corpus de textes dans sa globalité [Good, 1958]. La distinction de groupes thématiques de documents dans ce contexte permet l'exploration de l'ensemble de la collection de documents, ce qui peut être bénéfique lorsque l'on ne sait pas exactement ce que l'on recherche, lorsque l'on navigue au gré des informations

Application	Données concernées	Bénéfice	Exemple
Partitionnement du corpus	Corpus	Permettre l'exploration du corpus	[McKeown *et al.*, 2002]
Clustering de résultats	Résultats d'une recherche préliminaire	Faciliter l'interprétation des résultats	[Zamir and Etzioni, 1999]
Recherche de groupes de documents	Corpus	Accélérer la recherche	[Salton, 1971a]
Modélisation du langage	Corpus	Améliorer l'ordre des résultats	[Liu and Croft, 2004]

TABLE 2.1 – Applications des méthodes de clustering en recherche d'information

recueillies pour récolter des informations diverses sur un large sujet [Manning *et al.*, 2008]. Par exemple, ce type d'application des techniques de clustering s'avère bien adapté à l'exploration de collections contenant des articles d'actualités : on n'a pas de but précis, l'on cherche à se renseigner sur ce qui s'est passé dernièrement dans le monde et le fait d'utiliser un système regroupant les articles par catégories permet une perception facilitée des différents évènements. *Google News* [Google, 2008] et son précurseur, le *Columbia NewsBlaster System* [McKeown *et al.*, 2002], sont des exemples de systèmes utilisant des techniques de clustering de manière statique. Le fait que, selon la *Cluster Hypothesis*, les documents d'un même groupe soit susceptibles d'être pertinents aux même requêtes permet à l'utilisateur de satisfaire un besoin général par l'exploration d'un cluster unique.

Cependant, lorsque le besoin d'information est plus spécifique, ce type d'utilisation du clustering s'avère peu performant. En effet, le nombre de documents pris en compte par le processus ne permet pas une distinction très fine des différents groupes thématiques. Une alternative possible est alors de réaliser une recherche d'information préliminaire en considérant la requête de l'utilisateur et d'appliquer le processus de clustering sur le sous-ensemble ainsi obtenu [Preece, 1973]. Ici, l'objectif est d'améliorer l'interprétation des résultats, de réduire les efforts cognitifs que l'utilisateur doit fournir pour localiser les informations qu'il recherche. Non seulement la catégorisation des résultats d'une recherche d'information classique permet d'orienter l'utilisateur vers les documents pertinents plus rapidement mais cela peut aussi le renseigner sur la diversité des informations du corpus en rapport avec son sujet [2]. Par ailleurs, si la *Cluster Hypothesis* se vérifie sur le corpus de textes utilisé, l'on peut supposer que l'ensemble des documents pertinents à la requête se retrouvent dans le même cluster et que l'utilisateur n'a alors que ce groupe à explorer pour satisfaire ses besoins. De très nombreuses approches se sont intéressées à ce type d'application des techniques de clustering [Leuski, 2001a; Tombros, 2002], conduisant à la mise à disposition de systèmes grand public tels que

2. Bien qu'il faille encore que les différents groupes thématiques soient représentatifs des différents aspects de la requête. Nous discuterons de ce point au chapitre 8.

le système *Grouper* [Zamir and Etzioni, 1999] cité dans le chapitre précédent, ou le très populaire système *Vivìsimo* [Koshman *et al.*, 2006].

Les techniques de clustering peuvent également servir à accélérer le processus de recherche d'information. Il peut en effet être bénéfique, plutôt que de comparer la requête de l'utilisateur à chaque document individuellement, de réaliser un premier filtrage selon la proximité des différents groupes thématiques produits préalablement sur l'ensemble des documents de la collection utilisée. La comparaison de la requête avec les documents individuels peut alors se faire sur le sous-ensemble correspondant au groupe le plus proche du besoin de l'utilisateur uniquement (qui est susceptible de contenir l'ensemble des documents pertinents si la *Cluster Hypothesis* se vérifie sur le corpus de textes utilisé), ce qui peut éviter de réaliser un grand nombre de calculs [Salton, 1971a]. Pour des systèmes basés par exemple sur le modèle vectoriel, dans lequel la recherche des documents les plus proches de la requête peut se faire de manière très rapide, ce filtrage par sélection du meilleur groupe n'apporte pas d'amélioration significative du temps de réponse (et provoque par ailleurs une baisse de la qualité des résultats car certains documents pertinents peuvent n'être pas contenus par le groupe choisi). Néanmoins, lors de l'utilisation de systèmes basés sur d'autres modèles, tel que le modèle LSI (voir section 1.3.1), ne permettant pas une interprétation aussi efficace de l'index inversé, l'utilisation de groupes thématiques pré-formés peut permettre d'obtenir des résultats bien plus rapidement que si la recherche avait été réalisée sur la collection entière.

La quatrième et dernière application mentionnée dans la table 2.1 concerne la modélisation de langage. On cherche ici à améliorer la recherche elle même (améliorer l'ordre des documents dans la liste produite). Le principe est de s'appuyer sur un partitionnement de la collection de documents pour améliorer le modèle de langue qui est à la base de nombreux systèmes récents. Il a été vu en section 1.3.1 sur les modèles de langue que, bien souvent, un modèle de la collection est utilisé pour "lisser" le modèle du document produit. Or, la collection est susceptible de contenir de nombreux documents qui n'ont strictement rien à voir avec le document en question. Le fait d'utiliser plutôt un modèle du groupe du document pour lisser le modèle du document peut permettre d'améliorer les performances de la recherche [Liu and Croft, 2004] puisque, selon la *Cluster Hypothesis*, le lissage est alors réalisé selon des documents qui tendent à être pertinents aux mêmes requêtes que le document concerné.

Notons qu'en plus de ces quatre applications, les méthodes de clustering ont été appliquées aux termes des documents [Doyle, 1964] pour étendre des requêtes [Jones, 1971; Van Rijsbergen *et al.*, 1981], construire [Crouch and Yang, 1992] ou mettre en relation des thésaurus [Amba *et al.*, 1996]. Néanmoins, ces applications, qui ne s'appuient pas sur la *Cluster Hypothesis*, sortent quelque peu du cadre de cette thèse. Nous nous limiterons dans la suite de ce chapitre à l'étude des méthodes de clustering appliquées aux documents, et plus particulièrement à la deuxième application de la table 2.1, c'est à dire le clustering des résultats retournés par une

recherche d'information préliminaire.

2.2 Relations entre documents

La première question à se poser lorsque l'on cherche à réaliser la mesure de relations thématiques pour produire un clustering d'un ensemble donné de n documents concerne la manière de représenter les textes, tel que c'est le cas lorsque l'on cherche à déterminer la relation entre un document et une requête. Quel type d'unités indexer ? Quel modèle de représentation utiliser ? Comment pondérer les termes ? Dans le modèle vectoriel, les documents sont représentés dans un espace vectoriel à n_t dimensions (avec n_t le nombre de termes différents dans le corpus, c'est à dire le nombre d'entrées de l'index inversé) en utilisant des mots simples comme unités d'indexation [Tombros, 2002]. Grâce à sa simplicité de mise en place et ses performances démontrées, ce type de représentation est le plus employé dans la littérature [3]. C'est alors celle que nous empruntons pour nos expérimentations. L'ensemble des termes d'indexation (les mots non vides des documents) sont utilisés pour estimer les similarités thématiques entre documents [4].

Contrairement à ce qui est communément accepté par la majorité de la communauté, Sneath et Sokal [Sneath and Sokal, 1973] ont longtemps conseillé, pour des raisons de simplicité, l'utilisation de vecteurs binaires dans le cas d'estimations de similarité thématique entre documents, pensant qu'une pondération des termes ne permet pas d'améliorer suffisamment la représentation pour se reporter significativement sur la qualité du clustering résultant. Salton et Buckley [Salton and Buckley, 1988] contredisent cette affirmation en présentant des résultats de clustering bien supérieurs lors de l'utilisation d'une pondération des termes selon une formulation en $tf * idf$. Des expérimentations ont par ailleurs montré dans [Wise, 1999] que les termes rares, lorsqu'ils ne sont pas uniques, peuvent s'avérer être les meilleurs indices de catégorisation. La prise en compte de ces termes avec une considération de leur fort pouvoir discriminant (idf) est alors importante.

Une fois la représentation des documents établie, il s'agit de définir une mesure des relations thématiques existant entre les documents. Un grand nombre de mesures ont été proposées dans la littérature pour mesurer la proximité ou la distance entre objets [Ellis *et al.*, 1993]. Sneath et Sokal [Sneath and Sokal, 1973] distinguent quatre types de mesures : les mesures d'association (ou de similarité), les mesures de dissimilarité, les mesures probabilistes et les mesures de corrélation. L'utilisation

3. Notons que Hatzivassiloglou et al. [Hatzivassiloglou *et al.*, 2000] ont obtenu des résultats légèrement supérieurs en augmentant leur représentation par utilisation de phrases comme unités d'indexation. Néanmoins, la rapidité d'exécution en est largement affectée.

4. Shaw [Shaw, 1993] a émis de nombreux doutes concernant l'efficacité de mesures considérant l'ensemble des termes d'indexation, montrant que la méthode Single-Link (voir section 2.3.3) obtenait de moins bons résultats lorsque tous les termes étaient considérés que lorsque l'on fixait un seuil de fréquence des termes dans le document (voir section 1.2.2). Néanmoins, les expériences de Burgin [Burgin, 1995] ont montré que ce phénomène n'est observé qu'avec cette méthode de clustering particulière.

des deux derniers types de mesures restant marginale, nous nous limitons à l'étude des deux premiers, qui ne font généralement qu'un puisqu'avec des mesures de similarité comme la mesure Cosine (section 1.3.2), dont le score est compris entre 0 et 1, la dissimilarité $\delta(D_i, D_j)$ entre deux documents D_i et D_j s'obtient en prenant simplement le complément à 1 de leur similarité $Sim(D_i, D_j)$. En reprenant la formule 1.27 à laquelle l'ensemble des mesures du modèle vectoriel peuvent être associées, on a alors :

$$\delta(D_i, D_j) = 1 - (Sim(D_i, D_j)) = 1 - \sum_{t \in D_i \cap D_j} W_{D_i,t} \times W_{D_j,t} \tag{2.4}$$

Dans [Willett, 1983], Willett a démontré l'importance d'utiliser une mesure qui réalise une normalisation des scores en fonction de la longueur des documents. Il paraît en effet naturel d'éviter que les longs documents se voient attribuer une espérance de similarité plus importante que les petits en raison de leur nombre de termes. Les comparaisons entre mesures réalisées dans [Willett, 1983] et [Rorvig, 1999] ont fait apparaître que la mesure Cosine permet de réaliser des clusterings de documents d'une qualité légèrement supérieure aux autres mesures utilisant une normalisation des scores selon la longueur des textes. C'est donc la mesure que nous employons dans les expérimentations que nous présentons dans les chapitres suivants. Le poids que nous utilisons pour chaque terme t dans un document D_i est donné par :

$$W_{D_i,t} = \frac{tf_{D_i,t} \times idf_t}{\sqrt{\sum_{t' \in D_i} (tf_{D_i,t'} \times idf_{t'})^2}} \tag{2.5}$$

Cette formulation permet de considérer à la fois l'importance des termes dans les documents (tf) ainsi que leur pouvoir discriminant dans le corpus (idf), tel que cela est réalisé dans la plupart des systèmes proposant un clustering des documents [5].

2.3 Méthodes de clustering

De très nombreuses méthodes de clustering, dont Berkhin donne un bon aperçu dans [Berkhin, 2006], ont été proposées dans la littérature, chacune présentant des caractéristiques propres qui la rendent plus ou moins adaptée à telle ou telle application. On distingue les méthodes hiérarchiques, dont le principe est de créer un arbre dans lequel chaque nœud correspond à un cluster regroupant deux clusters de plus bas niveau pour fournir une hiérarchie représentative de la structure du jeu de données concerné (figure 2.2), des méthodes non-hiérarchiques (ou *Flat Clustering*) qui ne produisent qu'un seul niveau de clusters.

5. Notons qu'en plus de ces deux facteurs, certaines approches incluent une considération de la requête dans le calcul de similarité inter-documents. Nous discuterons de ce point plus en détail au chapitre 8.

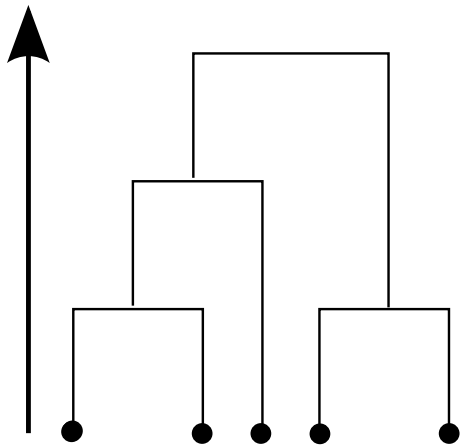

FIGURE 2.2 – Exemple de hiérarchie

Le type de clustering le plus employé en recherche d'information est certainement le clustering hiérarchique. Le principal avantage de ce type de clustering réside dans le fait qu'il n'implique pas la détermination d'un nombre de clusters *a priori* [Willett, 1988]. La hiérarchie produite propose des partitions de tout niveau, le choix d'une partition spécifique de l'ensemble de documents peut, si besoin est, se faire *a posteriori* selon le dendogramme défini. Par ailleurs, les méthodes hiérarchiques semblent plus stables que les méthodes non-hiérarchiques [Van Rijsbergen, 1979].

Les méthodes non-hiérarchiques requièrent bien souvent l'instanciation d'un grand nombre de paramètres (nombre de clusters, taille des clusters, critères d'optimisation, etc...). Dans le cadre d'un clustering des résultats d'une recherche d'information préliminaire, ce paramètrage peut s'avérer difficile à réaliser puisque les valeurs définies doivent dépendre de l'ensemble de documents retournés par le système de recherche initial. Par ailleurs, une limite bien connue de ces méthodes est que leur partition finale dépend souvent de la partition initiale et de l'ordre dans lequel les documents ont été examinés [Bergmark *et al.*, 1977]. Néanmoins, ces méthodes présentent elles aussi leurs avantages : les partitions obtenues sont généralement de meilleure qualité que les partitions que l'on peut extraire d'une hiérarchie pour un nombre de groupes donné. Cela s'explique par le fait que, contrairement à celle obtenue par une méthode non-hiérarchique, la partition extraite de la hiérarchie n'a pas été optimisée par des réaffectations d'objets aux groupes [Tufféry, 2005]. Par ailleurs, les méthodes non-hiérarchiques sont généralement bien plus rapides que les méthodes hiérarchiques [Rocchio, 1966].

L'ensemble des méthodes de clustering utilisent des relations de distances entre objets ou entre objets et groupes d'objets. Néanmoins, si le calcul de distances entre documents est maintenant bien défini (formule 2.4), le mode de représentation des clusters reste à déterminer pour être capable d'estimer la distance entre un document

et un cluster ou la distance entre deux clusters. Avant de présenter quelques unes des méthodes de clustering (hiérarchiques et non-hiérarchiques[6]) couramment utilisées en recherche d'information, nous explorons les différents modes de représentation des clusters qu'il est possible de mettre en place.

2.3.1 Mode de représentation des clusters

Afin d'être capable de mesurer des distances inter-clusters $\delta(C_i, C_j)$ ou la distance entre un document et un cluster $\delta(D_i, C_j)$, un mode de représentation des clusters doit être défini. Il s'agit de déterminer un vecteur de poids de termes représentatif du contenu du cluster afin de pouvoir le comparer aux vecteurs des documents ou des autres clusters (ou même au vecteur d'une requête dans le cas d'une recherche de clusters pertinents [Salton, 1971a], voir la troisième application de la table 2.1, ou d'une présentation ordonnée des différentes catégories produites). Deux facteurs jouent un rôle déterminant dans la qualité de la représentation d'un cluster [Tombros, 2002] :

- Le vecteur représentatif doit rendre compte de l'essentiel de l'information contenue par les documents du cluster concerné,
- Le vecteur représentatif doit pouvoir être distingué des autres clusters (pouvoir discriminant du vecteur).

Il faut alors que le représentant du groupe soit choisi de manière à synthétiser l'information que le cluster contient, tout en étant suffisamment discriminant, c'est à dire suffisamment éloigné des autres clusters, afin de souligner ses caractéristiques propres.

Un grand nombre d'approches ont été proposées pour permettre une représentation des clusters efficace [Croft, 1978; Van Rijsbergen, 1979]. Typiquement, le vecteur représentatif (ou noyau) d'un cluster est une combinaison linéaire des vecteurs des éléments contenus par ce cluster [Diday *et al.*, 1982]. Salton a introduit dans [Salton, 1971b] la notion de centroïde (ou centre de gravité d'un cluster) qui correspond au vecteur dont les poids sont égaux à la moyenne des poids des termes correspondants dans l'ensemble des vecteurs de documents du cluster concerné. Le poids $W_{C,t}$ du terme t dans le vecteur représentatif $\vec{C}$ du cluster C s'obtient alors par :

$$W_{C,t} = \sum_{D_i \in C} \frac{W_{D_i,t}}{|C|} \tag{2.6}$$

où $|C|$ est le nombre de documents contenus par le cluster C. Bien qu'un tel représentant soit au centre du cluster, ce mode de représentation ne s'avère que rarement efficace puisqu'il ne permet pas de rendre suffisamment compte des différences

6. Notons que des approches hybrides, mêlant ces deux types de clustering, ont été proposées, notamment l'approche introduite dans [Bellot, 2000] qui combine hiérarchie et méthode de nuées dynamiques ou celle décrite dans [Lebart *et al.*, 2000] qui allie un algorithme *K-Means* à une méthode hiérarchique ascendante.

d'un cluster donné par rapport aux autres. De trop nombreuses caractéristiques sont souvent partagées par les différents clusters. De nombreuses variantes ont alors été proposées pour améliorer la définition d'un vecteur représentatif performant. Par exemple, Murray [Murray, 1972] a proposé de supprimer du vecteur représentatif les termes dont le nombre d'occurrences dans le cluster, ou plutôt dans les documents du cluster, n'est pas assez important. Cette approche, supprimant certains termes partagés par plusieurs clusters des vecteurs représentatifs, permet généralement une légère amélioration du mode représentation [7]. D'autres approches proposent de travailler avec des vecteurs binaires plutôt qu'avec des vecteurs pondérés. Par exemple, Jardine et Van Rijsbergen [Jardine and Van Rijsbergen, 1971] représentent les clusters par des vecteurs où $W_{C,t} = 1$ signifie la présence d'un terme dans un nombre de documents supérieurs à un seuil donné (la valeur de seuil préconisée est $log_2|C|$). Cela permet de ne conserver que les caractéristiques communes à plusieurs documents du cluster.

Alternativement à la production de vecteurs représentatifs artificiels, il est possible de représenter chaque cluster par la sélection d'un ou de plusieurs de leurs éléments [Diday *et al.*, 1982]. De nombreuses possibilités de sélection se présentent alors. Lors d'une mesure de distance d'un élément avec un cluster, il est par exemple possible d'utiliser l'élément du cluster le plus proche de l'élément externe tel que cela est fait dans la méthode hiérarchique *Single Link* [Van Rijsbergen, 1979]. À l'inverse, à l'instar de la méthode *Complete Link*, il est possible de sélectionner l'élément le plus éloigné comme représentant du cluster. La méthode *Group Average*, quant à elle, ne détermine pas de représentant particulier mais définit la distance entre un élément et un cluster comme la moyenne des distances entre cet élément et l'ensemble des éléments du cluster. Dans [Bellot, 2000], une autre représentation considérant plusieurs éléments de chaque cluster est utilisée : les clusters sont représentés par leurs k éléments centraux, c'est à dire les k éléments dont la somme des distances aux autres éléments du cluster est minimale. Enfin, il est envisageable d'utiliser le document le plus discriminant du cluster, c'est à dire le plus éloigné des documents des autres clusters [Gelbukh *et al.*, 2003]. Dans ce dernier cas, les représentants permettront une bonne distinction des différents clusters. Ils risquent cependant de ne pas refléter suffisamment les informations des différents documents des groupes qu'ils représentent.

Tel que le souligne Tombros dans [Tombros, 2002], peu d'études comparatives entre les différents modes de représentation ont été réalisées. Il est alors difficile d'affirmer avec certitude quelle approche est la plus efficace. La question du meilleur mode de représentation des clusters reste ouverte.

7. Voorhees observe que la valeur seuil (fréquence minimale pour conserver un terme donné), et donc la taille du vecteur représentatif, a un impact considérable sur la qualité du clustering final obtenu, quelle que soit la méthode utilisée. Elle affirme alors que le contrôle de la taille du vecteur représentatif est un problème délicat qui nécessite une exploration approfondie.

2.3.2 Méthodes non-hériachiques

L'idée centrale de la plupart des méthodes de clustering non-hériachiques consiste à choisir une partition initiale de l'ensemble de données (ici des documents) que l'on transforme itérativement en réalisant des réaffectations d'appartenance aux groupes tant que de telles opérations permettent d'optimiser un ou plusieurs critères donnés [Anderberg, 1973]. Le nombre de partitions possibles de n documents en k clusters rend l'espace de recherche trop grand pour pouvoir tester toutes les solutions [Willett, 1988]. Les méthodes de clustering non-hiérarchiques ont alors recours à des heuristiques plus ou moins évoluées pour approximer la solution optimale. Nous détaillons ici quelques méthodes de clustering classiques [8].

Méthode Single-Pass

Algorithme 2.1 : Algorithme Single-Pass

Données :
Un ensemble $\mathcal{D}$ de n documents $D_1, D_2, \ldots, D_n$,
Une valeur seuil φ.
Résultat :
Une partition $\mathcal{C}$ de k clusters $C_1, C_2, \ldots, C_k$.

```
1  début
2  |  C_1 = ∅ ; C = {C_1} ; k = 1 ; cur = 2;
3  |  /* Initialisation du premier cluster */
4  |  C_1 = C_1 ∪ {D_1};
5  |  tant que cur ≤ n faire
6  |  |  si δ(D_cur, C_k) > φ alors
7  |  |  |  /* Création d'un nouveau cluster*/
8  |  |  |  k = k + 1 ; C_k = ∅ ; C = C ∪ {C_k};
9  |  |  fin
10 |  |  /* Ajout au cluster courant*/
11 |  |  C_k = C_k ∪ {D_cur};
12 |  |  cur = cur + 1;
13 |  fin
14 fin
```

Certainement l'algorithme de clustering le plus rapide de tous, l'algorithme *Single-Pass* [Frakes and Baeza-Yates, 1992] ne nécessite qu'un seul examen de chaque

8. Il ne s'agit bien sûr que d'une partie infime de la variété des méthodes existantes qui est présentée ici. Parmi les méthodes les plus connues que nous n'abordons pas figure notamment la méthode *EM* (*Expectation Maximisation*) [Dempster *et al.*, 1977] qui partitionne un ensemble d'objets en recherchant le modèle qui a pu générer ces données (en utilisant un critère probabiliste, le *Maximum Likelihood Criterion*, pour définir les paramètres du modèle). Une description détaillée des nombreuses méthodes de clustering non-hiéarchiques existantes a été récemment réalisée dans [Berkhin, 2006].

document pour décider de son affectation dans un cluster donné. Le principe est très simple : on considère les documents les uns après les autres pour décider si ils doivent être affectés au cluster courant où si l'on doit créer une nouveau groupe pour les y insérer. Pour chaque document D_i, cette décision est prise en fonction de sa distance $\delta(D_i, C_j)$ au cluster courant C_j et d'une valeur seuil maximale φ. L'algorithme 2.1 décrit le processus en détail.

Tel que le font remarquer Zamir et Etzioni, cette méthode est loin d'être optimale puisque la partition obtenue dépend très fortement de l'ordre dans lequel les documents sont considérés [Zamir and Etzioni, 1998].

Méthode des Nuées Dynamiques

Algorithme 2.2 : Algorithme des Nuées Dynamiques

Données :
Un ensemble $\mathcal{D}$ de n documents $D_1, D_2, \ldots, D_n$,
Une partition initiale $\mathcal{C}$ de l'ensemble $\mathcal{D}$ en k clusters.
Résultat :
Une partition $\mathcal{C}$ de k clusters $C_1, C_2, \ldots, C_k$.

```
1  début
2      C' = ensemble de k clusters C'_1, C'_2, ..., C'_k vides;
3      stop = 0;
4      best = 0;
5      tant que stop = 0 faire
6          stop = 1;
7          /* Ajout de chaque document au cluster le plus proche*/
8          pour i de 1 à n faire
9              best = argmin_{j∈{1,...,k}} δ(D_i, C_j);
10             C'_best = C'_best ∪ {D_i};
11         fin
12         si C ≠ C' alors
13             C = C';
14             C' = ensemble de k clusters C'_1, C'_2, ..., C'_k vides;
15             stop = 0;
16         fin
17     fin
18 fin
```

La méthode de clustering par *Nuées Dynamiques* [Diday *et al.*, 1982] est un exemple de méthode dite de "ré-allocation". Contrairement à la méthode *Single Pass* où, l'on ne réalise qu'un seul passage (soit une seule affectation par document), la méthode des *Nuées Dynamiques* relance le processus d'allocation tant que des transferts de documents d'un cluster dans un autre sont observés. Tel que décrit par

l'algorithme 2.2, la méthode vise à raffiner une partition initiale obtenue par une méthode de clustering externe en réaffectant les documents à des clusters différents si cela permet une amélioration de la cohésion des groupes. Il a été démontré dans [Diday *et al.*, 1982] que l'algorithme converge toujours vers une partition stable.

Un avantage non négligeable de cette méthode par rapport à la méthode *Single-Pass* réside dans le fait que la partition obtenue est indépendante de l'ordre dans lequel les documents ont été considérés. De plus, une certaine stabilité de la partition produite a été observée : l'ajout d'un nouveau document dans un cluster juste avant la convergence de l'algorithme ne remet pas nécessairement en cause la partition produite [Rasmussen, 1992]. Enfin, il est à noter que la méthode n'est que peu sensible à la métrique utilisée pour calculer les distances entre objets [Schütze and Silverstein, 1997].

Méthode K-Means

Certainement la plus célèbre des méthodes de clustering, la méthode *K-Means* [Tou and Gonzalez, 1974; Kaufman and Rousseeuw, 2005], cherche à produire une partition d'un ensemble de données qui minimise la distance des objets avec le centre du cluster qui les contient. L'objectif est de trouver l'ensemble optimal de centres de clusters.

La méthode commence par choisir aléatoirement k (pour k clusters) points dans l'espace vectoriel[9]. Ces points représentent des prototypes de clusters et le but est alors de les attirer vers les centres des clusters optimaux. Tant que le processus (*cf.*, algorithme 2.3) n'a pas atteint une partition stable, la méthode affecte les objets (dans notre cas les documents) au cluster de leur plus proche prototype pour former les nouveaux groupes dont on calcule les centres qui constitueront les nouveaux points prototypes de la prochaine itération. Il a été observé à maintes reprises que l'algorithme converge rapidement vers une partition satisfaisante dans la plupart des cas [Kaufman and Rousseeuw, 2005]. Du fait de sa simplicité et ses performances, cet algorithme a été employé de manière intensive durant ces trente dernières années [Manning *et al.*, 2008].

Une des principales difficultés que pose l'utilisation de la méthode *K-means* est la détermination du nombre k de clusters à produire. Néanmoins, de nombreuses approches, telles que la méthode *AIC* (Akaike information criterion) présentée dans [Akaike, 1974] ou la *Gap Statistic* établie dans [Tibshirani *et al.*, 2000], proposent une estimation automatique de ce nombre. De nombreuses variantes de cette méthode existent, notamment la méthode *fuzzy C-means* [Bezdek, 1981] qui en est une version *Soft* (affectation possible d'un même élément à plusieurs clusters), la méthode *QT* (*Quality Threshold*) [Heyer *et al.*, 1999] qui n'impose pas de spécifier un nombre de clusters mais se base sur la taille des clusters ou la méthode *K-Medoïd* [Kaufman

9. Notons que ces points peuvent aussi correspondre aux centres de clusters produits par une méthode externe.

Algorithme 2.3 : Algorithme K-Means

Données :
Un ensemble $\mathcal{D}$ de n documents $D_1, D_2, \ldots, D_n$,
Un ensemble $\mathcal{P}$ de k points $P_1, P_2, \ldots, P_k$ dont les coordonnées correspondent à des poids de termes $W_{P_i,t}$.
Résultat :
Une partition $\mathcal{C}$ de k clusters $C_1, C_2, \ldots, C_k$.

```
début
    C = ensemble de k clusters C_1, C_2, ..., C_k vides;
    C' = ensemble de k clusters C'_1, C'_2, ..., C'_k vides;
    stop = 0 ; best = 0;
    tant que stop = 0 faire
        stop = 1;
        /* Ajout de chaque document au cluster
        correspondant à leur prototype le plus proche*/
        pour i de 1 à n faire
            best = argmin_{j∈{1,...,k}} δ(D_i, P_j);
            C'_best = C'_best ∪ {D_i};
        fin
        si C ≠ C' alors
            C = C';
            C' = ensemble de k clusters C'_1, C'_2, ..., C'_k vides;
            stop = 0;
            / * Calcul des nouveaux centres */
            pour i de 1 à k faire
                pour tous les termes t ∈ ⋃_{D_j∈D} faire
                    W_{P_i,t} = 1/|C_i| × Σ_{D_j∈C_i} W_{D_j,t};
                fin
            fin
        fin
    fin
fin
```

and Rousseeuw, 1990] dont les prototypes associés aux clusters sont des éléments de l'ensemble à partitionner (les medoïdes) plutôt que des points de l'espace vectoriel (ou centroïdes).

2.3.3 Méthodes hiérarchiques

Les méthodes hiérarchiques ne génèrent pas une partition unique de l'espace de données, mais produisent un empilement de partitions de différents niveaux, souvent représentées sous la forme d'un dendogramme (figure 2.2) qui décrit les

Algorithme 2.4 : Algorithme de clustering hiérarchique ascendant

Données :
Un ensemble $\mathcal{D}$ de n documents $D_1, D_2, \ldots, D_n$,
Une valeur seuil maximale φ.
Résultat :
Une hiérarchie $\mathcal{H}$ de $n-1$ partitions $H_1, H_2, \ldots, H_{n-1}$.

début
  $\mathcal{H} = \emptyset$; $\mathcal{C} = \emptyset$;
  $stop = 0$; $k = n$; $A = 0$; $B = 0$;
  /* Initialisation des clusters */
  pour *i de* 1 *à* n **faire**
    $C_i = \{D_i\}$;
    $\mathcal{C} = \mathcal{C} \cup \{C_i\}$;
  fin
  tant que $stop = 0$ **faire**
    $stop = 1$;
    $H_{n-k+1} = \mathcal{C}$;
    $\mathcal{H} = \mathcal{H} \cup \{H_{n-k+1}\}$;
    /* Recherche des clusters les plus proches*/
    $(A, B) = \text{argmin}_{(i,j)\in\{1,\ldots,k\}^2, i\neq j}\, \delta(C_i, C_j)$;
    si $\delta(C_i, C_j) \leq \varphi$ **alors**
      $stop = 0$;
      $C_A = C_A \cup C_B$;
      $\mathcal{C} = \mathcal{C} \setminus \{C_B\}$;
      Supprimer les références à C_B dans la matrice des distances;
      Modifier les distances entre chaque cluster et le cluster C_A;
      $k = k - 1$;
      Renuméroter les clusters de $\mathcal{C}$ de 1 à k;
    fin
  fin
fin

transitions entre partitions successives. Ces méthodes peuvent être ascendantes (agglomératives) ou descendantes (divisives) [Mirkin, 1996]. Alors que dans le premier cas le principe est de réaliser des fusions successives de clusters, dans le second l'objectif est de scinder des groupes pour obtenir des clusters plus petits. Dans les deux cas, le processus utilise des métriques de distance entre les groupes. À chaque itération, les méthodes ascendantes cherchent alors à réaliser la fusion de groupes qui nuira le moins à la cohésion moyenne des clusters et les méthodes descendantes tentent de trouver la scission de groupes qui pénalisera le moins le degré de dissimilarité moyenne entre clusters. Les méthodes ascendantes sont largement préférées en recherche d'information [Van Rijsbergen, 1979;

Willett, 1988]. Nous limiterons donc notre étude des méthodes hiérarchiques à ces seules approches agglomératives.

L'algorithme 2.4 décrit le processus général des méthodes hiérarchiques ascendantes. Au début du processus de clustering, chaque objet est assigné à un cluster différent que l'on va chercher à fusionner avec d'autres pour obtenir au bout du compte un seul groupe contenant la totalité des objets de l'ensemble initial (dans le cas où l'on ne fixe pas de seuil permettant d'arrêter le processus) [10]. À chaque itération du processus, l'algorithme regroupe les deux clusters d'objets les moins distants et modifie la matrice de dissimilarités en conséquence. Les différences entre les multiples méthodes agglomératives résident dans l'heuristique de modification de cette matrice. En effet, tel que nous l'avons vu en section 2.3.1, de nombreuses représentations des groupes sont possibles. Des différences de représentations conduisent à des calculs de distance différents et par là même, à des regroupements de clusters différents. Selon la méthode utilisée, la distance entre deux clusters correspond à :

- *Single Link :* La plus petite distance entre objets des deux clusters,
- *Complete Link :* La distance maximale entre objets des deux clusters,
- *Weighted Group Average :* La distance moyenne entre éléments des deux clusters,
- *Group Average :* La distance moyenne entre éléments des deux clusters normalisée par la taille des groupes concernés,
- *Centroid :* La distance entre les centres des deux clusters,
- *Ward :* La somme totale des carrés des distances des éléments au point central de chaque cluster.

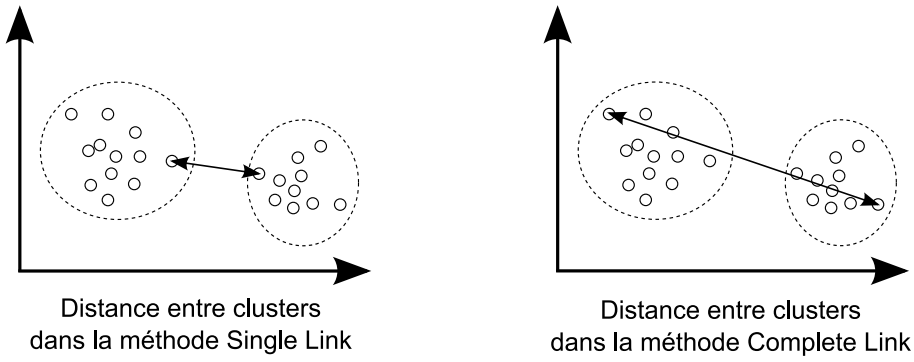

FIGURE 2.3 – Distances entre clusters dans les méthodes *Single Link* et *Complete Link*

10. L'extraction d'une partition de la hiérarchie peut alors être souhaitée selon le type d'application concernée. De manière imagée, il est communément admis que, en considérant la forme du dendogramme résultant de la méthode de clustering, la hiérarchie doit être coupée au niveau où "les collines deviennent montagnes" [Beck, 2006]. Ceci revient à extraire une partition de l'arbre au moment où les éléments restant à fusionner commencent à être fortement dissimilaire.

Méthode	α_i	α_j	β	γ_i
Single Link	0.5	0.5	0	-0.5
Complete Link	0.5	0.5	0	0.5
Weighted Group Average	0.5	0.5	0	0
Group Average	$\frac{\lvert C_i\rvert}{\lvert C_i\rvert+\lvert C_j\rvert}$	$\frac{\lvert C_j\rvert}{\lvert C_i\rvert+\lvert C_j\rvert}$	0	0
Centroid	$\frac{\lvert C_i\rvert}{\lvert C_i\rvert+\lvert C_j\rvert}$	$\frac{\lvert C_j\rvert}{\lvert C_i\rvert+\lvert C_j\rvert}$	$\frac{-\lvert C_i\rvert\times\lvert C_j\rvert}{(\lvert C_i\rvert+\lvert C_j\rvert)^2}$	0
Ward	$\frac{\lvert C_i\rvert+\lvert C_l\rvert}{\lvert C_i\rvert+\lvert C_j\rvert+\lvert C_l\rvert}$	$\frac{\lvert C_j\rvert+\lvert C_l\rvert}{\lvert C_i\rvert+\lvert C_j\rvert+\lvert C_l\rvert}$	$\frac{-\lvert C_l\rvert}{\lvert C_i\rvert+\lvert C_j\rvert+\lvert C_l\rvert}$	0

TABLE 2.2 – Coefficients de Lance et Williams

Afin de simplifier et accélérer les calculs à réaliser pour modifier la matrice de distances entre clusters après chaque fusion de groupes, Lance et Williams ont montré que, pour l'ensemble de ces méthodes, la nouvelle distance entre un cluster donné C_l et le cluster $C_i \cup C_j$ résultant de la fusion des groupes C_i et C_j pouvait s'obtenir en utilisant l'équation suivante [Lance and Williams, 1967] :

$$\delta_{C_l,C_i\cup C_j} = \alpha_i \times \delta_{C_l,C_i} + \alpha_j \times \delta_{C_l,C_j} + \beta \times \delta_{C_i,C_j} + \gamma \times |\delta_{C_l,C_i} - \delta_{C_l,C_j}| \qquad (2.7)$$

où α_i, α_j, β et γ sont des coefficients donnés dans la table 2.2 selon la méthode utilisée. L'utilisation de cette équation de généralisation des méthodes agglomératives permet de réduire considérablement le temps requis pour transformer la matrice de distances initiale.

De nombreuses études comparatives de ces différentes méthodes de clustering ont été réalisées [Willett, 1988]. Il a été observé que la méthode *Single Link* tend à produire des groupes fortement dissimilaires mais de faible cohésion interne alors que la méthode *Complete Link* tend plutôt à produire des groupes peu séparés les uns des autres mais dont les éléments sont relativement proches [Griffiths *et al.*, 1997]. Ces deux méthodes constituent ainsi les deux extrêmes de l'assertion stipulant que les clusters d'une partition doivent posséder une forte cohésion interne tout en étant bien séparés les uns des autres [Mirkin, 1996]. Les autres méthodes représentent des compromis entre ces deux extrêmes. Une comparaison approfondie de ces différentes méthodes a fait ressortir dans [Tombros, 2002] que la méthode *Group Average* permet la production de la hiérarchie la plus adaptée à la recherche d'information.

2.4 Description du contenu des groupes thématiques

L'application de techniques de clustering aux résultats d'une recherche d'information préliminaire permet bien souvent d'améliorer leur présentation et de réduire ainsi les efforts que l'utilisateur doit fournir pour trouver les informations qu'il recherche. Reste à déterminer le meilleur moyen de présenter les clusters obtenus à l'utilisateur. Dans la section 1.5, nous avons vu qu'il pouvait être profitable de

présenter des descriptions des catégories à l'utilisateur. Nous proposons ici d'explorer les différentes techniques d'extraction du contenu des groupes pour en produire une description.

La description du contenu des clusters peut se faire de multiples manières. Tel qu'aperçu en section 1.5, deux grands courants se distinguent :

– Extraction d'informations communes à l'ensemble des documents du groupe,
– Sélection d'un ou de plusieurs documents représentatifs de chaque groupe.

Choisir entre ces deux types de présentation des catégories de documents revient à se poser la question suivante : vaut-il mieux tenter de synthétiser les différentes informations abordées par l'ensemble des documents du groupe, avec les difficultés techniques que cela implique sachant que chaque document individuel peut lui même aborder plusieurs thématiques (voir le chapitre 7 pour une discussion sur le clustering de passages thématiques permettant d'obtenir des groupes mieux centrés autour d'une thématique spécifique), ou plutôt rechercher le ou les documents "leaders" du groupe, avec le risque non négligeable de sélectionner des documents peu représentatifs ou difficilement appréhendables ?

L'extraction d'informations des groupes pour en décrire le contenu peut se rapprocher du problème du choix d'un mode de représentation des clusters pour permettre des calculs de proximité entre groupes de documents (section 2.3.1). En effet, la plupart des approches proposées travaillent par repérage des termes les plus représentés dans le cluster ou par sélection de phrases en rapport avec les thèmes abordés par les documents du groupe [Anick and Vaithyanathan, 1997; Maarek *et al.*, 2000]. Néanmoins, bien que très proches, les objectifs de ces deux champs de recherche ne sont pas les mêmes. On ne cherche pas ici à déterminer une représentation interne du groupe mais à extraire de l'information interprétable par un utilisateur final. La difficulté peut alors sembler accrue puisqu'il faut prendre en compte des facteurs cognitifs mis en jeu lors de l'utilisation d'un système de recherche d'information. D'après Tombros [Tombros, 2002], les améliorations dans ce domaine seront probablement apportées par la communauté du résumé automatique, qui a montré ses capacités à traduire le degré de pertinence d'un document [Tombros and Sanderson, 1998], et plus particulièrement de celle du résumé multi-documents [Mani, 2001; Dang, 2005]. Il est en effet envisageable d'appliquer les approches proposées dans ce dernier domaine aux différents groupes thématiques de manière individuelle pour en faire ressortir les informations principales.

Alternativement, il est donc possible de décrire les groupes thématiques en y sélectionnant des documents représentatifs. Dans [Gelbukh *et al.*, 2003], trois différents modes de sélection d'un document représentatif dans les groupes sont proposés : le médoïde (i.e., le document situé le plus au centre du groupe), qui est susceptible de donner la meilleure idée des informations partagées par l'ensemble de documents du cluster, le document plus discriminant (i.e., le document le plus distant des documents des autres groupes), qui permet de souligner les différences ou caractéristiques spécifiques au cluster concerné, et le document le plus commun (i.e.,

le document le plus proche des documents des autres clusters), qui permet d'identifier des différences plus fines entre les groupes [11] (l'utilisation d'un tel représentant permet de garantir un niveau de distinction entre les groupes, les contenus de deux différents groupes ne seront pas plus proches que leurs représentants). Dans un contexte plus orienté vers le domaine de la recherche d'information, trois autres modes de sélection ont été proposés dans [Leuski, 2001a] : le mieux classé (i.e., le document le plus proche de la requête de l'utilisateur), qui donne la possibilité d'écarter les clusters déconnectés du besoin d'information (si ce document n'est pas pertinent alors le reste du groupe risque de ne pas l'être non plus), le moins bien classé (i.e., le document le plus éloigné de la requête), qui permet de garantir un degré de pertinence du groupe (si ce document est pertinent alors le reste du groupe a des chances de l'être aussi), et le document de rang moyen (i.e., le document dont le rang de similarité avec la requête correspond au rang médian des documents du cluster), qui peut s'avérer intéressant car il représente un bon compromis entre les deux extrêmes précédents [12]. À notre connaissance, il n'existe malheureusement pas d'étude comparative de ces différents modes de sélection dans un contexte de présentation des résultats.

Bien qu'un certain nombre d'études aient été réalisées pour comparer les différents modes de description du contenu des clusters [Barry, 1998; Kural, 1999], aucune d'entre elles n'a permis d'établir de manière certaine lequel d'entre eux permet la meilleure appréhension de leur contenu et qui plus est, dans une optique de recherche d'information, la meilleure localisation de l'information pertinente. D'après [Barry, 1998], il ressortirait que l'utilisation de la partie *abstract* (ou résumé) de documents représentatifs constitue le meilleur indicateur de pertinence des groupes, suivis par les résumés automatiques de ces documents représentatifs (utiles lorsque les documents représentatifs ne contiennent pas d'*abstract*), les titres des documents représentatifs, les listes de citations de ces documents puis enfin des termes d'indexation extraits des groupes en fonction de leur représentativité. Il apparaît alors que l'utilisation de documents représentatifs permet une meilleure description du contenu des documents que l'extraction des informations abordées par l'ensemble des documents des groupes, quel que soit le mode de présentation de ces documents représentatifs. Cette affirmation est néanmoins à prendre avec précaution puisque, tel que le fait remarquer Tombros [Tombros, 2002], l'étude citée a été réalisée dans un cadre restreint, où la recherche d'information concernait la localisation d'un document unique, ce qui ne reflète pas la majorité des besoins d'information réels. Par ailleurs, les évolutions techniques réalisées depuis 1998 peuvent avoir modifié l'ordre établi [Barry, 1998].

11. Ces deux derniers types de représentant peuvent être rapprochés des représentants utilisés dans les méthode hiérarchique *Complete Link* et *Single Link*.

12. D'après Lewis [Lewis, 1992], s'intéresser aux documents moyennement bien classés est le meilleur moyen de trouver la séparation entre les informations pertinentes et non-pertinentes.

2.5 Conclusion

La justification principale de l'utilisation de techniques de clustering en recherche d'information repose sur la *Cluster Hypothesis*, qui stipule que les documents pertinents ont tendance à être plus similaires les uns avec les autres qu'avec les documents non pertinents. La plupart des approches qui utilisent une catégorisation des résultats visent alors à former un groupe contenant l'ensemble des documents pertinents. Ainsi, si la *Cluster Hypothesis* se vérifie sur le corpus et la requête considérés, et que la méthode de clustering établie permet un bon regroupement des documents les plus similaires, il suffira à l'utilisateur d'identifier le groupe qui lui semble le plus correspondre à ses attentes pour localiser les informations qu'il recherche. Les bénéfices d'une telle application des techniques de clustering ne se limitent alors pas à rendre la présentation des résultats d'une recherche d'information plus attractive. Regrouper les documents selon leur proximité thématique peut aussi permettre à des documents pertinents ne contenant que peu d'occurrences des termes de la requête, et risquant alors de ne pas être examinés dans le cadre d'une présentation par liste ordonnée de documents, d'être affectés au groupe des documents pertinents et d'avoir alors la possibilité d'être identifiés comme tels. Leuski [Leuski, 2001b] fait remarquer que ce type d'application du clustering peut s'avérer au moins aussi efficace que la majorité des techniques d'expansion de requêtes par réinjection de pertinence, tout en étant moins contraignant pour l'utilisateur.

Néanmoins, si une telle présentation paraît permettre une interprétation des résultats facilitée, elle n'est pas sans soulever quelques interrogations : qu'advient-il si la requête comporte plusieurs aspects clairement déconnectés ? La méthode forme-t-elle un groupe qui les englobe tous ? Dans ce cas, comment distinguer la structure de l'information pertinente ? Comment peut-on être sûr que des documents importants n'aient pas été affectés à des groupes autres que le groupe contenant la majorité des documents pertinents ? Ces différentes interrogations, dont nous rediscutons au chapitre 8, sont à la base d'une grande partie du travail présenté dans cete thèse.

Chapitre 3

Évaluation des systèmes de recherche d'information

Parce que toute proposition de système ou approche doit comporter un certain nombre d'expérimentations pour en déterminer les apports, nous présentons dans ce chapitre les protocoles et mesures d'évaluation qui sont couramment mis en œuvre dans le domaine de la recherche d'information. Dans une première section nous tentons d'en cerner les enjeux et problématiques afin d'appréhender les concepts proposés par la suite en gardant à l'esprit la complexité de la tâche d'évaluation. Alors qu'une seconde section décrit les modes de constitution des ressources d'évaluation, la troisième et dernière section de ce chapitre présente quelques méthodologies proposées pour estimer les performances des systèmes, que ceux-ci présentent à l'utilisateur une liste ordonnée de documents ou bien un ensemble de catégories thématiques de résultats.

Sommaire

3.1 Enjeux et problématiques 68
3.2 Corpus, requêtes et pertinence 69
3.3 Méthodologies d'évaluation 73
3.3.1 Liste ordonnée de résultats 73
3.3.2 Groupes de documents 76
3.4 Conclusion 81

3.1 Enjeux et problématiques

Quel que soit le domaine de recherche, la conception et le développement de modèles et méthodes pour réaliser une tâche donnée passe par des phases d'expérimentation pour en évaluer les performances. L'enjeu est alors de taille car il s'agit de définir des critères permettant de déterminer les apports de telle ou telle approche par rapport à telle autre. L'existence de biais expérimentaux, dûs par exemple à des critères d'évaluation mal adaptés ou tendant à favoriser un type d'approche particulier, peut conduire à orienter des recherches futures dans une mauvaise direction. Une mauvaise interprétation des résultats peut conduire à des conclusions erronées.

L'évaluation des systèmes de recherche d'information n'échappe pas à ces problématiques. Bien au contraire, alors que dans certains domaines la détermination de critères d'évaluation est relativement évidente - on sait exactement ce à quoi on s'attend et il suffit de comparer la sortie des méthodes avec la solution bien définie que l'on souhaite atteindre -, les considérations sémantiques des informations textuelles et l'aspect subjectif des résultats accroissent les difficultés liées à la mise en œuvre de méthodologies d'évaluation des approches. Bien que la meilleure méthode pour comparer des systèmes reste la considération d'appréciations humaines, la mise en place d'une telle évaluation peut s'avérer difficile. En effet, outre la difficulté de trouver un nombre suffisant de personnes prêtes à participer à l'évaluation, il faut mettre en place un protocole adéquat permettant la mise en évidence de tendances significatives [1]. Ainsi, une grande attention doit être portée aux informations données aux participants, aux objectifs qu'on leur fixe, aux types de résultats collectés, etc ... De très nombreux facteurs entrent en ligne de compte lors de la mise en œuvre de méthodologies d'évaluation, et ce d'autant plus si celles-ci impliquent une participation humaine. Sans compter que ce type d'évaluation ne peut être répété trop fréquemment du fait du nombre de personnes impliquées. Au vu de ces considérations, en dehors des grandes campagnes d'évaluation présentées dans la section suivante, la plupart des travaux en recherche d'information utilisent plutôt des méthodologies d'évaluation automatiques, qui permettent de réaliser des expérimentations sans requérir l'implication de sujets humains.

Si l'objectif des systèmes de recherche d'information est d'aider un utilisateur à localiser les informations susceptibles de répondre à ses besoins, la première question à se poser concerne la caractérisation de la notion de pertinence des informations. Selon Goffman [Goffman, 1968], la pertinence des informations est indissociable du besoin identifié. Elle dépend des connaissances que l'on a du sujet concerné et évolue au fur et à mesure que la recherche avance. Autrement dit, un document ne peut être déterminé comme pertinent sans considérer le contexte de la recherche d'information. Pire, la notion de pertinence évolue dans le temps, selon les informations collectées au cours de la recherche. Cette dernière observation s'avère très

1. Le lecteur intéressé pourra se référer à [Voorhees, 2002] ou [Harman, 1992a].

problématique puisqu'elle implique une prise en compte de la redondance de l'information fournie par les systèmes. Un document ne peut alors pas être jugé pertinent pour une requête donnée de manière statique, il faut prendre en compte ses apports en terme d'informations nouvelles. Ainsi, la pertinence du document dépend à la fois du besoin exprimé et des informations contenues par les autres documents de la collection. Néanmoins, face à la quasi-impossibilité de prendre en considération cette observation puisqu'il faudrait catégoriser les apports de chaque document de la collection utilisée, tâche qui risque d'introduire de nombreux biais puisque le clustering de données textuelles est, tel que nous l'avons vu au chapitre 2, une problématique à part entière (nécessitant elle-même la mise en place de critères d'évaluation), la plupart des campagnes d'évaluation des systèmes de recherche d'information font abstraction de l'aspect évolutif de la pertinence au cours de la recherche, considérant alors un besoin constant pour une recherche donnée, quelles que soient les informations contenues par les différents documents de la collection et l'ordre de présentation des documents en relation avec le sujet concerné. Cette simplification de la notion de pertinence permet la détermination de relations pérennes entre documents et requêtes : il est alors possible de définir un critère $Pert(D, R)$ dont la valeur dépend du degré de pertinence du document D en fonction de la requête R.

Bien que grandement facilité par l'abstraction du caractère évolutif de la pertinence, le problème de la détermination des données expérimentales n'est pour autant pas résolu. Se pose alors le problème de la détermination du degré de pertinence, auquel s'ajoute celui de la collecte de documents et de l'établissement de requêtes adaptées, dans le contexte de la tâche pour laquelle les approches évaluées ont été conçues. L'ensemble de ces problématiques, ainsi que la mise en place de mesures d'évaluation efficaces et équitables, sont abordées dans la suite de ce chapitre.

3.2 Corpus, requêtes et pertinence

Afin de disposer de ressources pour comparer les approches et systèmes proposés, des campagnes d'évaluation des systèmes de recherche d'information ont été organisées depuis un peu plus d'une quinzaine d'année. La plus populaire d'entre d'elles, la campagne proposée par la conférence *TREC* (*Text REtrieval Conference*) [Harman, 1995], a été initiée aux États-Unis par la *DARPA* (*Defense Advanced Research Projects Agency*) et le *NIST* (*National Institute for Science and Technology*) [2]. Cette campagne d'évaluation met à disposition de nombreux corpus d'évaluation, requêtes et jugements de pertinence permettant d'expérimenter différents systèmes de recherche d'information. Ces ressources sont articulées autour de différentes tâches,

2. Notons également l'existence d'une campagne francophone importante, le projet *Amaryllis* [Landi *et al.*, 1998], organisée par l'*INIST* (*INstitut de l'Information Scientifique et Technique*).

dont la principale concerne la recherche documentaire classique[3]. Les corpus mis à disposition par *TREC* pour cette tâche sont composés pour la plupart d'articles de journaux ou de rapports officiels. Par exemple, le corpus *ZIFF* contient des articles spécialisés en informatique publiés par la compagnie *Ziff-Davis*, *FR* des extraits du registre fédéral des États Unis et *AP* et *WSJ* des articles de presse issus respectivement de l'*Associated Press* et du *Wall Street Journal*. Pour ce qui est de l'établissement des requêtes, des thèmes sont fournis à des participants qui rédigent les questions en fonction des documents qu'ils trouvent dans les corpus (afin de rédiger des requêtes qui trouvent des réponses pertinentes dans la collection). Les requêtes finales, tel qu'illustré par la figure 3.1, comportent trois champs principaux nommés *Title*, qui correspond à une courte liste de mots-clés, *Description*, qui correspond à une description un peu plus longue du sujet, et *Narrative*, qui détaille les différents aspects des informations que l'on attend. Une fois les corpus constitués et les requêtes définies, il faut établir des relations de pertinence entre documents et requête.

Title: Conflicting Policy

Description: Document will cite an instance in which the U.S. government propounds two conflicting or opposing policies.

Narrative: A relevant document will cite an instance in which the U.S. federal government propounds two policies which are in conflict with or opposition to each other, or seem at least hypocritical.

FIGURE 3.1 – Requête n 74 de TREC-1

Idéalement, afin de pouvoir comparer les systèmes de recherche d'information, il faudrait pouvoir disposer de ressources sur lesquelles appliquer des recherches prédéfinies dont on connait précisément la réponse optimale (listes ordonnée de résultats, groupes de documents formés, etc ...). De telles ressources sont difficiles à mettre en place car elles nécessitent une correspondance manuelle entre documents utilisés et requêtes préalablement définies. Or, pour être représentatifs des collections de documents dans lesquelles sont réalisées les recherches réelles, les corpus de texte utilisés pour l'évaluation des systèmes doivent contenir de très nombreux documents. Repérer manuellement dans de tels corpus les documents pertinents à telle ou telle requête pré-définie et établir la liste ordonnée de résultats idéale (ou les groupes idéaux ou autres selon ce que l'on cherche à évaluer) est alors une tâche qui peut s'avérer, si ce n'est impossible, tout du moins très fastidieuse. Pour

3. Les autres tâches varient suivant les années. Elles concernent par exemple les systèmes interactifs [Over, 1998], la recherche interlangue (requêtes et documents de langues différentes) [Schauble, 1998], la recherche sur de très gros corpus [Hawking *et al.*, 1999] ou la recherche de documents oraux [Garofolo *et al.*, 1999].

construire une liste de documents ordonnés selon leur degré de pertinence, il faudrait en effet, pour chaque requête établie, non seulement repérer parmi l'ensemble des documents du corpus lesquels d'entre eux contiennent de l'information pertinente, mais en outre comparer les documents sélectionnés pour déterminer une échelle de pertinence des informations. Outre les efforts qu'il faudrait fournir pour comparer les documents sélectionnés, la pertinence des informations étant une notion relativement floue dont les appréciations peuvent différer selon les sensibilités des sujets réalisant l'annotation du corpus, la détermination d'une telle échelle paraît peu réaliste. Par conséquent, la plupart des campagnes d'évaluation se contentent de réaliser un jugement binaire de la présence ou non d'informations pertinentes dans chaque document. Ainsi, un document est jugé comme pertinent pour une requête donnée ($Pert(D, R) = 1$) si il contient des informations en rapport avec le sujet recherché, quel que soit le volume et la diversité de ces informations. Si cette simplification diminue le degré de finesse de l'évaluation, elle en améliore la fiabilité (puisque les divergences entre estimations de pertinence sont moindres [4]).

Dans *TREC*, afin de permettre une annotation du corpus dans un temps raisonnable, les relations de pertinence (qui sont donc binaires) sont construites à partir des résultats fournis par les systèmes participants à la campagne d'évaluation plutôt que sur la base des corpus entiers. Cette technique, appelée méthode de *Pooling* [Sparck-Jones and van Rijsbergen, 1975], vise à réduire le nombre de documents à examiner pour déterminer la liste de documents pertinents à chaque requête en ne considérant que les n (ici 100) premiers documents retournés par chaque système évalué lors de la campagne. Grâce à un tel procédé, les ressources humaines nécessaires à la détermination d'ensembles de documents pertinents sont évidemment moins importantes que celles requises pour examiner tous les documents des corpus. Il est à noter cependant que des biais expérimentaux peuvent être introduits par l'utilisation d'une telle méthode, puisque ne peuvent être identifiés comme pertinents que des documents rapportés par au moins l'un des systèmes expérimentés. Bien que de nombreux systèmes participent généralement aux campagnes *TREC* (environ 50 par an), de nombreux documents contenant de l'information pertinente risquent de ne pas être examinés [Zobel, 1998] [5]. Tel qu'observé dans [Buckley *et al.*, 2006], les documents pertinents qui ne comportent que peu de termes de la requête ne figurent que rarement dans les ensembles de documents pertinents établis lors des campagnes d'évaluation utilisant la méthode de *Pooling* (ce qui peut être pénalisant lorsque l'on parvient à retourner ces documents). De plus, l'utilisation d'ensembles de documents pertinents issus d'une telle méthode de *Pooling* ne permet pas nécessairement d'identifier les

4. Notons tout de même que, malgré cette simplification, Harman a observé que seulement 70 à 80% des individus s'accordent sur l'appréciation de la pertinence ou non d'un document [Harman, 1994].

5. Pour affirmer que de très nombreux documents pertinents n'ont pas été identifiés comme tels dans les campagne *TREC*, Zobel a cherché à estimer le nombre réel de documents pertinents dans le corpus en réalisant une extrapolation des différences entre les nombres de documents pertinents identifiés lorsque l'on considère divers nombres k de premiers documents retournés par l'ensemble des systèmes participants.

Corpus	ZIFF	AP	WSJ	FR
Nombre de documents	75180	84510	98732	25960
Termes par document	297	217	204	927
Termes uniques par document	139	144	128	244
Termes des 1000 documents les + courts	19	14	11	49
Termes des 1000 documents les + longs	7915	512	1235	11304
Termes uniques des 1000 documents les + courts	14	13	12	35
Termes uniques des 1000 documents les + longs	1503	323	623	1392
Documents pertinents par requête	54.06	40.44	54.26	19.32

TABLE 3.1 – Statistiques des collections de documents

apports novateurs potentiels d'une méthode donnée [de Loupy and Bellot, 2000; de Loupy, 2000]. En effet, si un système retourne un document contenant de l'information pertinente n'ayant pas été classé dans les 100 premiers documents des listes produites par les systèmes participants à la campagne de *TREC*, il ne sera pas pris en compte par l'évaluation, ou plutôt il sera considéré comme non pertinent, puisqu'il n'appartient pas à l'ensemble de documents examinés lors du repérage réalisé. Néanmoins, sous réserve que les systèmes ayant participé à la campagne d'évaluation soient relativement performants, l'utilisation d'ensembles de documents ainsi constitués permet tout de même d'évaluer les systèmes "hors compétition" (c'est à dire hors du cadre de la campagne d'évaluation) relativement efficacement. Ce qui est alors évalué est la capacité du nouveau système à retrouver les documents pertinents retournés par au moins un des systèmes participants à la campagne d'évaluation. Par ailleurs, notons que si un système n'ayant pas participé à l'obtention de l'ensemble de documents pertinents obtient de bons résultats par rapport à des systèmes y ayant pris part, il est possible d'augurer d'encore meilleurs résultats lors d'une participation à une nouvelle campagne d'évaluation. Nous reviendrons sur les biais relatifs à la méthode de *Pooling* au chapitre 6.

La table 3.1 présente les statistiques des quatre corpus principaux utilisés tout au long de nos expérimentations. Dans cette table, les nombres de termes donnés correspondent aux nombres moyens de mots non vides dans chaque document et les nombres de termes uniques sont obtenus en ne conservant qu'une seule occurrence de chaque terme dans chaque document. On peut noter que chaque corpus contient une relativement grande variété de tailles de documents. Alors que *FR* contient un grand nombre de très longs documents, *AP* regroupe quant à lui des documents relativement courts. *ZIFF* et *FR* présentent la plus grande variété de tailles de documents. Le même jeu de requêtes, les topics 1-50 de *TREC*, est utilisé pour les différents corpus. Dans les expérimentations présentées dans la suite de ce rapport, nous utilisons deux formes des mêmes requêtes (voir figure 3.1) : la partie *Title* des requêtes, qui contient 2.95 termes en moyenne (soit 2.95 uniques), et la partie

Narrative, qui contient une moyenne de 88.08 termes (soit 52.6 uniques).

3.3 Méthodologies d'évaluation

Une fois que l'on dispose d'une collection et d'un jeu de requêtes pour lesquelles des ensembles de documents pertinents ont été constitués, il convient de s'interroger sur les mesures d'évaluation pour comparer les systèmes de manière adéquate. Cette évaluation, qui ne peut qu'être relative et dépendante de l'application visée, doit permettre la production de résultats représentatifs des performances réelles des systèmes. Nous explorons dans cette section les différentes mesures proposées pour deux types de systèmes de recherche d'information différents, les systèmes classiques présentant une liste ordonnée des résultats et les systèmes proposant une catégorisation thématique des documents.

3.3.1 Liste ordonnée de résultats

La liste ordonnée de résultats, proposée par les systèmes classiques, peut être évaluée de diverses manières. Dans tous les cas néanmoins, ce que l'on cherche à considérer est la mesure dans laquelle les documents pertinents sont classés en tête de liste, puisque cela détermine en quelque sorte la capacité de l'utilisateur à trouver les informations qui l'intéressent.

Traditionnellement, plutôt que se s'intéresser à la totalité de la liste produite, on ne s'intéresse qu'à un nombre donné de ses premiers documents que l'on appelle les documents "retournés" par le système. Ceci permet de définir, pour une recherche effectuée, 4 sous-ensembles de documents : les documents pertinents retournés, les documents non pertinents retournés, les documents pertinents non retournés et les documents non pertinents non retournés. Les deux principaux critères d'évaluation utilisés en recherche d'information classique, le *Rappel* et la *Précision*, rendent comptent de la cardinalité de deux de ces sous-ensembles par rapport aux autres, celui des documents pertinents retournés et celui des documents non pertinents non retournés. Plus formellement, si $\mathcal{P}ert$ correspond à l'ensemble des documents pertinents et $\mathcal{R}et$ à celui des documents retournés, on peut définir ces deux critères comme suit [Van Rijsbergen, 1979] :

$$Rappel = \frac{|\mathcal{P}ert \cap \mathcal{R}et|}{|\mathcal{P}ert|} \tag{3.1}$$

$$Precision = \frac{|\mathcal{P}ert \cap \mathcal{R}et|}{|\mathcal{R}et|} \tag{3.2}$$

Ainsi, alors que le rappel constitue la proportion de documents pertinents retournés par le système, la précision rend compte du ratio de documents pertinents dans l'ensemble des documents retournés par le système.

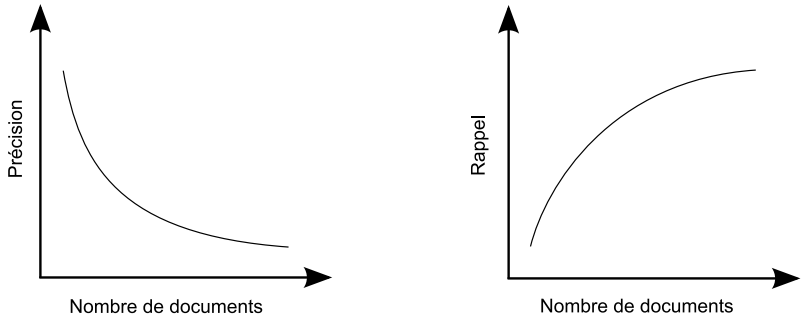

FIGURE 3.2 – Précision et rappel selon le nombre de documents considérés.

Les graphes de la figure 3.2 illustrent les tendances générales observées de l'influence du nombre $n = |\mathcal{R}et|$ de documents retournés sur les deux critères de précision (graphe de gauche) et de rappel (graphe de gauche) [6]. Alors que le rappel augmente avec le nombre de documents retournés (la probabilité de trouver des documents pertinents étant plus élevée lorsque le nombre de documents considéré grand), la précision diminue dans le même temps (la probabilité de retourner des documents non pertinents augmentant d'autant que le nombre de documents retournés est grand). Ainsi, lorsque l'on cherche à augmenter la précision d'un système, c'est très souvent au détriment de la qualité de son rappel et réciproquement [Van Rijsbergen, 1979]. L'objectif est de réaliser un compromis optimal entre ces deux critères.

De manière à considérer les deux critères sur un même graphe, il est possible d'interpoler les valeurs de précision correspondant à différents niveaux de rappel. Généralement, la règle d'interpolation utilisée pour construire un tel graphe définit la précision pour un niveau de rappel i comme étant la valeur maximale de précision pour tout niveau de rappel supérieur ou égal à i. Par exemple, s'il existe 100 documents pertinents pour une requête, la valeur interpolée de la précision pour un rappel de $0,1$ correspond à la meilleure précision obtenue avec au moins 10 documents pertinents. La figure 3.3 représente trois courbes "*Rappel / Précision*" correspondant aux interpolations des valeurs de précision selon les valeurs de rappel pour trois listes de documents ordonnées. Alors que la courbe 1 de ce graphe correspond à des performances clairement inférieures à celles décrites par les courbes 2 et 3, la comparaison entre ces deux dernières courbes est bien plus délicate. Tout dépend de l'utilisation que l'on veut faire des résultats. Souhaite-t-on obtenir de très nombreuses informations sur le sujet, quitte à ce que certains documents retournés soient de qualité médiocre, ou préfère-t-on récupérer un nombre restreint de documents, plus probablement bien ciblés sur le sujet de la recherche ? Par exemple, dans le cas d'une application à la veille technologique, afin de ne pas passer à côté d'infor-

6. Les notations $P@x$ et $R@x$ parfois utilisées dans la littérature correspondent aux valeurs de précision et de rappel à des niveaux donnés x (nombre de premiers documents considérés parmi les documents retournés), on parle de précision ou de rappel au rang x.

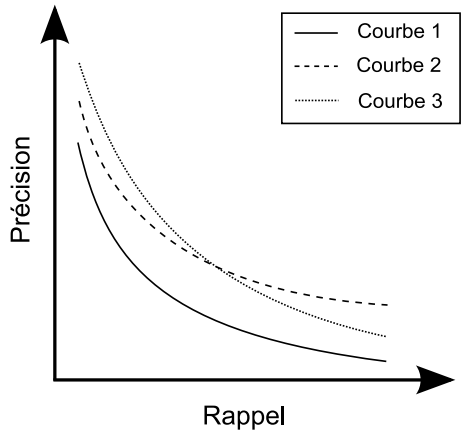

FIGURE 3.3 – Courbes Rappel / Précision

mations qui pourraient s'avérer importantes, on privilégiera le rappel (courbe 2), alors que pour une recherche correspondant à un besoin d'information plus précis, la précision sera préférée (courbe 3).

Il peut s'avérer néanmoins que l'on souhaite obtenir une valeur numérique unique correspondant à la qualité générale d'une liste ordonnée de documents. Ainsi de nombreux travaux utilisent la précision moyenne (*Average Precision*) qui rend compte des deux critères de rappel et de précision. Elle correspond à la moyenne des précisions calculées après chaque document pertinent de la liste ordonnée de résultats. Si $\mathcal{D} = \{D_1, \ldots, D_n\}$ représente la liste ordonnée des n premiers résultats retournés par le système [7] (D_1 correspond au premier document retourné) et $Pert(D_i)$ une fonction $Pert : \mathcal{D} \rightarrow \{0, 1\}$ retournant 1 si le document D_i est pertinent (0 sinon), la précision moyenne d'une liste $Ap(\mathcal{D})$ s'obtient par :

$$\begin{aligned} Ap(\mathcal{D}) &= \frac{1}{\sum_{i=1}^{n} Pert(D_i)} \times \sum_{i=1}^{n} Pert(D_i) \times P@i \\ &= \frac{1}{\sum_{i=1}^{n} Pert(D_i)} \times \sum_{i=1}^{n} \sum_{j=1}^{i} \frac{Pert(D_i) \times Pert(D_j)}{i} \end{aligned} \tag{3.3}$$

Cette mesure, qui nous paraît être la mesure d'évaluation de listes ordonnées la plus informative, permet de prendre en compte le rang des éléments pertinents dans la liste plutôt que de considérer uniquement le taux de documents pertinents dans

7. Généralement, on considère $n = Nb_D$, l'ensemble des documents du corpus. Dans l'éventualité où le système n'aurait pas retourné la totalité des documents, les documents restants sont ajoutés en fin de liste.

un ensemble [8]. Cette mesure est utilisée dans la plupart des évaluations de listes ordonnées de documents.

Parmi les autres mesures d'évaluation existantes figure la précision relative, qui correspond au score de précision mesuré après un nombre donné de documents pertinents. Cette mesure permet de comparer les listes de résultats en tenant compte du nombre de documents à trouver. L'ordre des réponses n'est alors pas considéré.

Enfin, notons que plusieurs combinaisons des critères de rappel et de précision ont été proposées pour obtenir une valeur unique. La plus répandue est celle proposée par Van Rijsbergen [Van Rijsbergen, 1979] qui suggère de combiner les valeurs de rappel et de précision selon l'importance accordée par l'utilisateur à l'un ou l'autre de ces deux critères. Cette mesure, appelée F-mesure, utilise alors un paramètre β permettant d'ajuster l'importance du rappel par rapport à la précision [9] :

$$F = \frac{(1 + \beta^2) \times Precision \times Rappel}{\beta^2 \times Precision + Rappel} \tag{3.4}$$

Un état de l'art sur les multiples mesures d'évaluation des listes ordonnées de documents est donné dans [Baeza-Yates and Ribeiro-Neto, 1999]. Les propriétés mathématiques des critères de rappel et de précision sont etudiées dans [Buckland and Gey, 1999].

3.3.2 Groupes de documents

Alors que l'évaluation des systèmes présentant une liste ordonnée de documents peut se faire en considérant les positions des documents pertinents dans cette liste, la caractérisation des performances des systèmes réalisant une catégorisation des documents peut s'avérer plus difficile. Bien sûr, il est possible de juger la qualité d'une méthode de clustering en considérant par exemple des critères de cohésion ou de séparation des groupes formés (formules 2.2 et 2.3 du chapitre 2) [10], mais ce qui nous intéresse surtout ici est l'accessibilité à l'information pertinente qu'offre le système évalué. Il s'agit alors d'évaluer la capacité qu'aura l'utilisateur à trouver les

8. Notons que, tel que démontré dans [Kishida, 2005], un tel calcul accorde une attention plus importante aux positions des premiers documents pertinents de la liste. Ceci n'est néanmoins pas nécessairement un biais puisqu'il paraît par exemple naturel de considérer le passage d'un document pertinent de la position 2 à la position 1 d'une liste plus important qu'un passage de la position 100 à la position 99.

9. Les trois valeurs les plus couramment utilisées pour ce paramètre sont $\beta = 0.5$, qui attribue plus d'importance à la précision qu'au rappel, $\beta = 1$, qui donne le même poids aux deux critères, et $\beta = 2$, qui permet au rappel de peser plus que la précision dans le score d'évaluation finalement obtenu. La F_1-mesure, couramment rencontrée dans la littérature, correspond à l'application de la F-mesure avec la valeur $\beta = 1$.

10. De nombreuses mesures permettent d'évaluer la qualité d'un partitionnement. Outre les mesures de cohésion et de séparation (évaluation interne qui peut par ailleurs se calculer de multiples façons, par exemple en considérant des distances aux représentants des groupes plutôt que de prendre en compte l'ensemble des distances entre éléments), des mesures d'évaluations externes (utilisation d'un ensemble de données dont on connait le partitionnement idéal) ou relatives (prise en compte de regroupements d'éléments sur lesquels s'accordent de nombreuses autres méthodes) peuvent être envisagées [Beck, 2006].

informations qui l'intéressent dans la partition présentée, ce qui semble plus complexe puisque de nombreux facteurs entrent en ligne de compte. Contrairement aux systèmes classiques où les documents sont présentés séquentiellement, les systèmes produisant des groupes de documents proposent un parcours des résultats plus complexe. On ne connait alors pas *a priori* l'ordre dans lequel les documents seront présentés à l'utilisateur, puisque cela dépend de ses choix de clusters successifs. Nous présentons dans cette section les différentes propositions qui ont été faites pour réaliser des évaluations automatiques dans un tel contexte.

Potentiel de validité de la Cluster Hypothesis

Tel que le soulignent de nombreuses études (voir par exemple [Tombros *et al.*, 2002]), il peut être utile de commencer l'évaluation d'un système par une étude de la matrice de similarités qu'il utilise pour déterminer sa propension à produire un partitionnement qui puisse être utile à la recherche d'information. Plus précisément, il s'agit de caractériser le potentiel de validité de la *Cluster Hypothesis* sur les documents concernés et les mesures de distances réalisées. On cherche alors à déterminer dans quelle mesure il sera possible de séparer les documents pertinents des non pertinents selon les relations inter-documents que le système a défini. Le degré de séparation entre documents pertinents et non pertinents est en effet un facteur important puisqu'il influe directement sur la capacité du système à finalement présenter un cluster contenant un bon ratio de documents pertinents.

Lorsqu'ils ont formulé la *Cluster Hypothesis*, Jardine et Van Rijsbergen ont établi un test pour en vérifier la validité. Ce test, nommé *Overlap Test* (test de recouvrement) [Jardine and Van Rijsbergen, 1971], évalue dans quelle proportion les documents pertinents sont plus similaires les uns avec les autres qu'avec les documents non pertinents. Pour chaque requête de l'évaluation, ce test s'intéresse alors aux relations de similarité existant entre les documents pertinents (pertinent-pertinent) et entre les documents pertinents et les documents non pertinents (pertinent-non pertinent) pour déterminer dans quelle mesure les deux distributions de similarités se recouvrent l'une l'autre. Plus le recouvrement est important, moins la *Cluster Hypothesis* est alors supposée se vérifier sur les documents et mesures de distances utilisées.

Bien que ce test soit directement dérivé de la *Cluster Hypothesis*, Voorhees a fait remarqué qu'il présentait un biais expérimental important [Voorhees, 1986] : le fait que le nombre de relations pertinent-non pertinent soit beaucoup plus important que le nombre de relations pertinent-pertinent cause des distorsions significatives dans les résultats obtenus. Par conséquent, elle a proposé un autre test, le test des plus proches voisins (*Nearest Neighbour Test*), pour évaluer la capacité du système à catégoriser efficacement les documents. Ce test consiste à compter le nombre de documents pertinents appartenant aux N (avec généralement $N = 5$) plus proches voisins, en terme de similarité, de chaque document pertinent de l'ensemble à partitionner. Plus le nombre de documents pertinents en voisinage direct est important,

plus il sera aisé pour le système de produire un cluster correspondant aux besoins de l'utilisateur.

Ces tests, qui ne réalisent pas une évaluation des résultats finaux des systèmes mais cherchent à caractériser leur potentiel de performances, permettent surtout de comparer différentes représentations des documents et mesures de similarités pouvant être utilisées par des systèmes de recherche d'information réalisant une catégorisation des résultats.

Reconstruction de listes ordonnées

Afin d'évaluer les systèmes réalisant un clustering des résultats de la même façon que les systèmes classiques (en utilisant par exemple la mesure de précision moyenne), certaines approches ont proposé de reconstruire des listes ordonnées de documents à partir des clusters formés [Hearst and Pedersen, 1996; Silverstein and Pedersen, 1997; Bellot and El-Bèze, 1999]. Pour ce faire, les approches commencent généralement par définir un ordre entre les clusters (couramment selon la proximité à la requête de leur document qui en est le plus proche ou bien la similarité moyenne de leurs documents à la requête [11]) et entre les documents de chaque groupe (selon leur similarité avec la requête ou avec le représentant du groupe) pour obtenir un ensemble ordonné $\mathcal{C} = \{C_1, \ldots, C_k\}$ de k sous-ensembles ordonnés $C_i = \{C_i^1, \ldots, C_i^{|C_i|}\}$ (C_i^j correspond alors au j-ième document du i-ième cluster C_i). Cette liste de listes (ou ensembles ordonnés) définit l'ensemble des chemins que l'utilisateur peut emprunter à travers les les groupes présentés : les clusters, représentés par leur document le plus proche de la requête, sont présentés par ordre de potentiel de pertinence et l'utilisateur les consulte pour sélectionner le groupe qui lui semble le plus adapté à son besoin. Les documents de ce cluster sont présentés par ordre de potentiel de pertinence ou de similarité avec le document représentant du groupe et l'utilisateur peut les examiner les uns après les autres (tel que c'est le cas avec la liste de résultats présentée par les systèmes classiques) jusqu'à avoir trouvé ce qu'il recherche ou décidé que les informations correspondant à ses besoins sont susceptibles d'être localisées dans un autre cluster, auquel cas il choisit un autre groupe à partir de la liste de représentants, examine les documents contenus dans ce groupe, etc... Les approches d'évaluation par reconstruction de listes ordonnées construisent leur liste finale $L = \{L_1, \ldots, L_n\}$ selon les chemins que l'utilisateur peut emprunter dans les sous-ensembles ordonnés C_i correspondant aux k différents clusters. La seule contrainte sur la construction d'une telle liste concerne l'ordre dans lequel les documents d'un cluster sont examinés : le document C_i^{j+1} ne peut être d'indice inférieur à C_i^j dans la liste finale L (puisque sauf exceptions, l'utilisateur examine les documents dans l'ordre où ils sont listés).

11. Notons que [Bellot and El-Bèze, 1999] proposent de considérer les jugements de pertinence établis plutôt que des estimations par mesures de similarité pour ordonner les clusters. Cela permet d'être certain de placer le cluster contenant l'ensemble des documents pertinents, s'il en est, en tête de liste. Les biais relatifs au choix d'un ordre entre les groupes sont alors quelque peu limités.

La figure 3.4 illustre deux parcours à travers les clusters, l'un valide, l'autre non valide. Les différents documents y sont représentés par des lettres (de A à

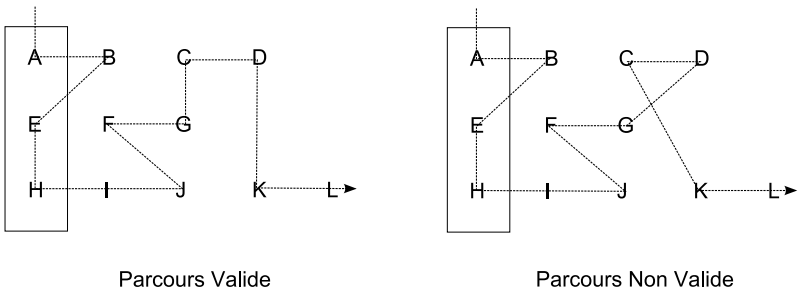

FIGURE 3.4 – Parcours à travers les clusters.

L) et les trois clusters proposés y sont disposés horizontalement. Les documents encadrés sont les représentants de groupe présentés initialement à l'utilisateur par ordre d'estimation de pertinence des clusters (de haut en bas). Les documents sont ordonnés à l'intérieur de chaque cluster (de gauche à droite) selon leur estimation de pertinence ou similarité avec le document représentant de leur groupe (A, E ou H). Le parcours de droite est invalide puisqu'il considère le quatrième document du premier cluster (D) avant le troisième (C), alors que l'utilisateur est supposé les examiner dans l'ordre.

Les différences entre les approches d'évaluation par reconstruction de listes ordonnées résident alors dans les parcours réalisés à travers les clusters. Les deux approches de reconstruction de listes les plus répandues s'inspirent des parcours souvent réalisés dans les arbres de recherche : le parcours en profondeur examine

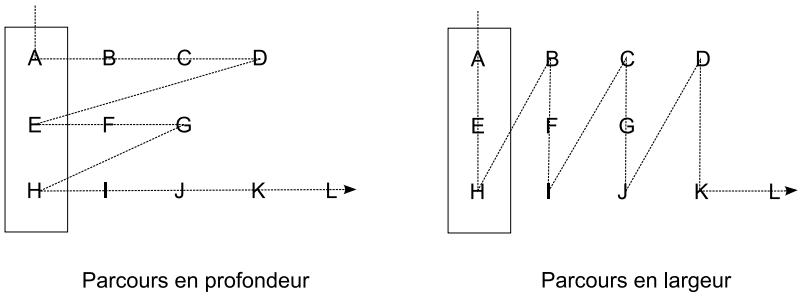

FIGURE 3.5 – Parcours en profondeur et parcours en largeur.

les clusters les uns après les autres en commençant par celui possédant le plus fort potentiel de pertinence, considérant alors la liste entière des documents de chaque cluster avant d'en changer, et le parcours en largeur considère séquentiellement les premiers documents non encore examinés de chaque liste, changeant de cluster après chaque examen de document. La figure 3.5 est une illustration de ces deux différents

parcours de listes. Ces deux parcours représentent les deux extrêmes de l'ensemble des chemins qu'un utilisateur peut emprunter. Ce que l'on évalue dans les deux cas est la facilité d'accès aux documents pertinents, déterminée par leur distribution et leur position dans les listes des clusters. Alors que l'évaluation d'une liste résultant d'un parcours en profondeur revient à considérer dans quelle mesure le système a réussi à regrouper l'ensemble de documents pertinents dans un même groupe, celle de la liste résultant d'un parcours en largeur traduit la répartition des documents pertinents dans les différents clusters et leur positionnement en tête de listes. Le parcours en profondeur paraît alors plus adapté si l'on considère que les systèmes réalisant une catégorisation des résultats s'appuient sur la *Cluster Hypothesis*. Le parcours en largeur peut néanmoins être intéressant et apporter des informations complémentaires sur la qualité des clusterings produits, par exemple dans le cas où la requête comporte plusieurs aspects bien distincts qui conduisent à la production de plusieurs clusters potentiellement intéressants. Nous reviendrons sur les approches d'évaluation par reconstruction de listes au chapitre 8.

Évaluation du cluster optimal

Considérant que la plupart des systèmes réalisant une catégorisation des résultats visent à regrouper l'ensemble des documents pertinents dans un même cluster, Jardine et Van Rijsbergen ont proposé d'évaluer les systèmes par une mesure du degré d'accomplissement de cette tâche. Ils ont ainsi introduit une approche d'évaluation qui consiste à juger la qualité d'un ensemble de groupes de documents (partition ou hiérarchie) en considérant uniquement le meilleur cluster qu'il contient. Le score attribué à un système par la mesure *MK1* proposée dépend alors du nombre et de la proportion de documents pertinents dans le meilleur cluster que l'on puisse trouver parmi l'ensemble $\mathcal{C}$ des clusters produits par le système [12] :

$$MK1 = \min_{C_i \in \mathcal{C}} \Big(1 - \frac{(1+\beta^2) \times Precision(C_i) \times Rappel(C_i)}{\beta^2 \times Precision(C_i) + Rappel(C_i)}\Big) \quad \text{avec}$$
$$Precision(C_i) = \frac{|\mathcal{P}ert \cap C_i|}{|C_i|} \quad \text{et} \quad Rappel(C_i) = \frac{|\mathcal{P}ert \cap C_i|}{|\mathcal{P}ert|} \tag{3.5}$$

où $Precision(C_i)$ correspond à la proportion de documents du cluster C_i à être pertinents et $Rappel(C_i)$ à la proportion des documents pertinents à figurer dans le cluster C_i [13]. Selon [Jardine and Van Rijsbergen, 1971] et [Tombros *et al.*, 2002], la

12. Ici, $\mathcal{C}$ correspond à l'ensemble des clusters produits par le système, que la méthode de clustering utilisée soit hiérarchique ou non. Dans le cas d'un clustering hiérarchique, $\mathcal{C}$ contient l'ensemble des clusters de la hiérarchie produite, quelqu'en soit le niveau.

13. On peut remarquer que la mesure de jugement de la qualité d'un cluster, la *E-mesure* [Jardine and Van Rijsbergen, 1971], est en fait le complément à 1 de la *F-mesure* présentée en section 3.3.1, à ceci près que, dans le contexte présent, les critères de précision et de rappel utilisés s'appliquent à un cluster et non à une liste ordonnée de documents.

mesure *MK1* présente l'avantage d'isoler la qualité du clustering des biais induits par l'utilisation d'une stratégie de recherche.

Une fois cette mesure établie, Jardine et Van Rijsbergen se sont interrogés sur la façon de l'utiliser pour comparer les performances de systèmes réalisant une catégorisation des résultats avec celles de systèmes de recherche classiques présentant une simple liste ordonnée de documents à l'utilisateur pour déterminer une fois pour toutes si l'application de techniques de clustering aux résultats d'une recherche d'information pouvait avoir un intérêt significatif [Jardine and Van Rijsbergen, 1971]. Il leur faut alors définir une mesure des performances des systèmes classiques qui puisse se comparer à la mesure *MK1*. La première idée fut, pour la comparaison d'un système réalisant une catégorisation des résultats avec un système classique, d'utiliser le nombre de documents du cluster considéré par la mesure *MK1* pour déterminer le nombre de documents à considérer lors de l'évaluation de la liste présentée par le système classique. Néanmoins, outre le fait qu'un tel procédé ne soit valide que lors de comparaisons de systèmes deux à deux, cela risque de pénaliser fortement le système classique puisque le nombre de documents à considérer est optimisé uniquement selon les groupes formés par le système réalisant une catégorisation des résultats. Par conséquent une autre mesure, nommée *MK3*, a été introduite dans [Jardine and Van Rijsbergen, 1971] pour réaliser des comparaisons de systèmes plus équitables. Cette mesure, suivant le même principe que la mesure *MK1* mais appliqué au contexte d'une liste de résultats, détermine un score d'évaluation qui dépend du meilleur groupe de premiers documents de la liste que l'on puisse trouver en faisant varier le nombre de documents considérés :

$$MK3 = \min_{i \in \{1,\dots,n\}} \left(1 - \frac{(1+\beta^2) \times P@i \times R@i}{\beta^2 \times P@i + R@i}\right) \tag{3.6}$$

avec $P@i$ et $R@i$ les scores de précision et de rappel au rang i de la liste de documents considérée.

L'établissement de ces deux mesures, toutes deux recherchant le meilleur groupe de documents que l'on puisse trouver selon la méthode utilisée, a permis d'observer que, dans la plupart des cas, l'application de techniques de clustering aux résultats d'une recherche d'information permet de présenter un groupe de documents bien plus intéressant pour l'utilisateur que la liste de résultats initiale [Jardine and Van Rijsbergen, 1971; Tombros *et al.*, 2002].

3.4 Conclusion

L'évaluation des systèmes de recherche d'information est un problème majeur qui occupe les chercheurs depuis l'apparition des premiers systèmes. Déjà dans [Kent *et al.*, 1955] étaient identifiés les principaux enjeux et problématiques auxquels se heurtent les expérimentations actuelles. Bien sûr, les outils et techniques ont depuis bien évolué mais le cœur du problème reste le même : comment juger de l'efficacité

d'un système alors que, pour un même besoin défini, les utilisateurs ne s'accordent que rarement sur ce qui est pertinent et ce qui ne l'est pas ? La pertinence est en effet une notion subjective complexe qu'il est très difficile de définir précisément. D'une part, elle varie selon les objectifs pour laquelle la recherche d'information est effectuée (résolution d'un problème ou recherche exploratoire, veille technologique ou recherche d'une information précise, compréhension d'un problème ou simple description, etc ...). D'autre part, elle dépend de la sensibilité, des connaissances, des capacités ou même ... des humeurs de l'utilisateur !

Dans un tel contexte, on est en droit de se demander comment il peut être possible de définir un protocole expérimental valide. Bien sûr, tel que nous l'avons vu, de nombreuses abstractions sont à réaliser, puisqu'il n'est pas raisonnable d'espérer considérer l'ensemble des facteurs qui régissent la recherche d'information, mais une fois simplifié, le problème de l'évaluation revient à la mise en place de critères permettant d'orienter les recherches vers des systèmes et modèles qui conviennent au plus grand nombre, qui puissent répondre à la majorité des besoins. C'est dans cette optique que de nombreux travaux se sont intéressés à l'établissement de méthodologies capables de rendre compte le plus justement possible des performances des systèmes. Des campagnes d'évaluation mettant en œuvre un grand volume de ressources humaines et techniques sont organisées ponctuellement pour faire le point sur les avancées dans le domaine de la recherche d'information. Les ressources mises à disposition par ces campagnes nous ont permis de mettre en place des expérimentations pour évaluer les performances des methodes que nous proposons dans la suite de ce rapport.

Chapitre 4

Meta-heuristiques et recherche d'information

Considérant les multiples facteurs entrant en jeu dans un processus de recherche d'information, la production d'un ensemble de textes répondant aux attentes d'un utilisateur peut constituer un problème requérant l'utilisation de techniques d'optimisation performantes. Ce chapitre présente quelques uns des principaux fondements de l'optimisation combinatoire, et plus particulièrement des méthodes de résolution approchée, en s'intéressant dans un premier temps à l'optimisation de problèmes à un seul objectif (problèmes mono-objectifs), pour terminer l'étude dans un contexte d'optimisation à objectifs multiples (problèmes multi-objectifs).

Sommaire

4.1 Optimisation combinatoire 84
4.2 Algorithmes génétiques 86
4.3 Optimisation multi-objectifs 90
4.3.1 La dominance au sens de *Pareto* 90
4.3.2 Le *Strength Pareto Evolutionary Algorithm* 93
4.4 Application à la recherche d'information 96
4.5 Conclusion 98

4.1 Optimisation combinatoire

De très nombreuses tâches de la vie courante correspondent, sans même que l'on ne s'en aperçoive, à des problèmes d'optimisation plus ou moins complexes. Ainsi, des tâches telles que ranger son bureau, placer son mobilier, établir un itinéraire ou choisir un appartement nécessitent l'établissement d'un ou de plusieurs objectifs à atteindre, la prise en compte des différents paramètres et contraintes du problème et la recherche de solutions permettant d'approcher le plus possible de ce que l'on considère comme une solution idéale. Alors que dans de nombreux cas, la modélisation du problème représente déjà un défi en soi, la détermination d'une solution optimale s'avère bien souvent constituer un objectif très difficile à atteindre.

L'optimisation combinatoire [Du and Pardalos, 1998], branche de recherche spécifique en informatique et en mathématiques appliquées, s'intéresse à la mise en œuvre de techniques et méthodes permettant la résolution de problèmes complexes de manière efficace. La tâche de résolution des problèmes d'optimisation consiste à trouver un ensemble de solutions qui optimisent un ou plusieurs critères selon les données du problème. Dans le cadre d'un problème de minimisation à un seul critère, on cherche à déterminer l'ensemble des solutions de l'espace réalisable $\mathcal{S}$ (c'est à dire, l'espace des configurations admissibles de l'espace de recherche selon les contraintes du problème) qui minimisent une fonction objectif f donnée :

$$\begin{array}{l} \text{Soit } f : \mathcal{S} \rightarrow \mathbb{R}, \\ \text{alors } \hat{\mathcal{S}} \text{ est l'ensemble des minimiseurs de } f \text{ tel que :} \\ \hat{\mathcal{S}} = \{x \in \mathcal{S} | \forall x' \in \mathcal{S}, f(x) \leq f(x')\} \end{array} \tag{4.1}$$

On distingue deux grandes familles de méthodes d'optimisation :

- Les méthodes exactes qui s'attachent à atteindre la ou les solutions optimales,
- Les méthodes approchées qui cherchent à atteindre des solutions représentant de bonnes approximations de la solution optimale.

La distinction entre ces deux familles de méthodes se fait surtout sur la garantie ou non d'obtenir la meilleure solution possible pour le problème donné. Alors que dans les méthodes exactes, la recherche de la solution optimale se fait généralement par énumération de l'ensemble des solutions de l'espace de recherche (avec ou sans utilisation d'heuristiques pour élaguer des branches de recherche), les méthodes approchées considèrent qu'une telle énumération est trop coûteuse et optent donc pour des techniques permettant d'orienter la recherche vers des solutions approchant l'objectif que l'on cherche à atteindre. Dans de nombreux problèmes qualifiés de

difficiles [1], la taille de l'espace de recherche rend en effet impossible l'énumération exhaustive de l'ensemble des configurations et l'on doit avoir recours à des algorithmes permettant d'approximer l'objectif à atteindre en un temps raisonnable. Sous la condition que l'optimalité de la solution n'est pas primordiale pour l'utilisation que l'on souhaite en faire, les méthodes approchées constituent une alternative très intéressante pour traiter les problèmes de grande taille. L'apparition de "meta-heuristiques" [2] [Glover and Kochenberger, 2003], il y a une quinzaine d'années, a permis d'affermir les performances des méthodes de résolution approchées [Reeves, 1993; Aarts and Lenstra, 1997], en proposant des schémas généraux permettant de restreindre l'exploration de l'espace de recherche. Ces meta-heuristiques sont essentiellement représentées par les approches de voisinage (méthode de *Descente*, méthode de *Recuit Simulé* [Kirkpatrick *et al.*, 1983], recherche *Tabou* [Glover, 1986], etc...), qui cherchent à approcher les solutions optimales du problème en "déplaçant" une configuration courante dans l'espace de recherche, et les approches bio-mémétiques (approches évolutionnaires [Jong and Spears, 1993] , algorithmes de colonies de fourmis [Dorigo and Caro, 1999], etc...), qui s'inspirent de phénomènes naturels pour orienter la recherche.

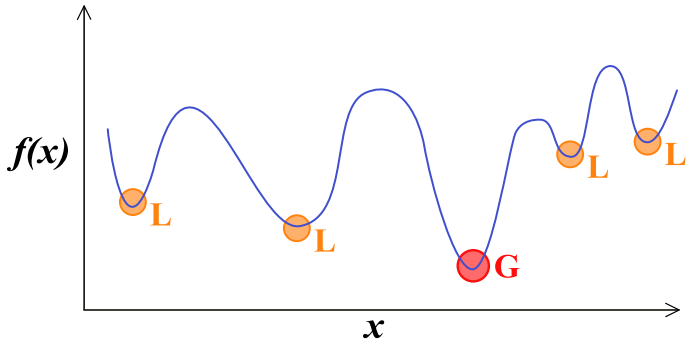

FIGURE 4.1 – Optimum global et optima locaux d'une fonction

La figure 4.1 illustre une recherche de l'ensemble des minimiseurs d'une fonction objectif f dans $\mathbb{R}$ (le problème représenté ne possède qu'une seule variable à optimiser). On y distingue un optimum global (point G), dont le score de la fonction

1. Parmi les classes de complexité établies figurent la classe P, qui regroupe l'ensemble des problèmes qui peuvent se résoudre en un temps polynômial à l'aide d'un algorithme déterministe, et la classe NP (*Non-Deterministic Polynomial*), qui contient l'ensemble des problèmes auxquels ne correspond pas d'algorithme déterministe de résolution en temps polynômial (mais peuvent se résoudre en un temps polynômial à l'aide d'algorithmes non-déterministes) [Papadimitriou, 1993]. Dans cette dernière classe NP, on retrouve notamment les problèmes *NP-difficiles* et les problèmes *NP-complets* dont la résolution implique l'utilisation de techniques d'optimisation performantes.

2. On parle d'heuristiques lorsque les techniques utilisées sont dédiées à l'optimisation d'un problème précis et de meta-heuristiques lorsque celles-ci peuvent être appliquées à une large variété de problèmes. Notons par ailleurs que le terme "meta-heuristique" a été initialement introduit dans [Glover, 1986] pour distinguer sa méthode Tabou des heuristiques spécifiques à des problèmes particuliers.

objectif est inférieur à celui de tous les autres points, et les différents optima locaux (points L), qui obtiennent un meilleur score que l'ensemble des configurations qui leur sont les plus proches. Bien que les modes de déplacement dans l'espace de recherche diffèrent selon le type de l'approche employée, l'existence de ces différents optima nous laisse entrevoir un problème majeur : il faut permettre au processus de recherche d'améliorer les solutions qu'il manipule, tout en lui donnant les moyens de ne pas se laisser enfermer dans des zones de configurations ne contenant pas l'optimum global du problème. Cela correspond à trouver un compromis entre intensification (déplacement vers le bas dans le cas de l'exemple de la figure 4.1) et diversification des configurations (déplacement de côté ou même vers le haut). C'est là qu'interviennent les techniques de meta-heuristiques qui visent à fixer un cadre permettant la mise en œuvre conjointe de telles opérations contradictoires.

De par l'adéquation du codage et des opérateurs de recherche qu'ils emploient, les algorithmes génétiques [Holland, 1975] nous sont apparus comme le type de meta-heuristiques le plus adapté aux problèmes que nous traitons dans cette thèse. La suite de ce chapitre se focalise donc sur ce type d'approches particulier, qui permet en outre de traiter des problèmes à objectifs multiples (voir section 4.3).

4.2 Algorithmes génétiques

La famille des approches évolutionnaires englobe l'ensemble des méthodes basées sur des concepts inspirés de l'évolution naturelle des espèces vivantes [Jong and Spears, 1993; Fonseca and Fleming, 1995]. Bien qu'à l'origine, ces approches n'étaient pas nécessairement destinées à la résolution de problèmes d'optimisation [Jong, 1992], elles sont maintenant fréquemment utilisées pour cette tâche, et constituent des alternatives très performantes aux approches de voisinage évoquées précédemment.

Selon les grands principes de l'évolution des espèces et de la sélection naturelle [Darwin, 1859], l'évolution d'une population est caractérisée par le degré d'adaptation des individus à leur environnement : en favorisant la survie et la reproduction des individus les mieux adaptés à leur milieu, la nature assure la pérennité des meilleures caractéristiques de la population et permet ainsi, par la recombinaison de ces caractéristiques entre elles, de faire évoluer l'espèce en formant des nouveaux individus qui tendent, au fil des générations, à être toujours mieux adaptés à l'environnement qui les entoure (puisqu'héritant des bonnes caractéristiques de leurs deux parents). Considérant la qualité d'une solution (selon le critère à optimiser) comme un degré d'adaptation au milieu, les approches évolutionnaires s'appuient sur ces principes pour obtenir des solutions proches de l'optimum du problème. Typiquement, un algorithme évolutionnaire est alors composé de trois éléments essentiels :

- Une population constituée d'individus représentant des solutions potentielles (configurations) du problème à optimiser ;
- Un fonction d'évaluation de l'adaptation (fonction de *fitness*) des individus à

leur environnement (objectif du problème) ;
- Un mécanisme d'évolution permettant la reproduction des individus et l'application du phénomène de sélection naturelle.

D'un point de vue opérationnel, un algorithme évolutionnaire typique fait évoluer une population en suivant un cycle composé de 3 étapes séquentielles :

1. Mesure de la qualité des individus de la population ;
2. Sélection d'une partie de la population ;
3. Production et/ou mutations d'individus ;

Sous réserve que le problème soit bien codé et que les opérateurs utilisés soient bien adaptés, la qualité des individus de la population devrait alors tendre à s'améliorer au fur et à mesure de ce processus d'évolution.

Les différentes approches évolutionnaires peuvent se classer en 3 grandes catégories :
- Les stratégies d'évolution [Schwefel, 1981] qui font évoluer une population en utilisant des opérateurs de sélection et de mutation déterministes ;
- La programmation évolutionnaire [Fogel *et al.*, 1966] qui ne réalise pas de sélection mais fait évoluer la population en appliquant des mutations successives aux individus ;
- Les algorithmes génétiques [Holland, 1975] qui, contrairement aux deux autres types d'approches, utilisent un opérateur de croisement des individus.

Ces trois types d'approches se différencient surtout par leur manière de représenter les données et de faire évoluer les populations de solutions qu'elles manipulent. Nous nous limitons ici à la présentation des algorithmes génétiques, qui sont les algorithmes évolutionnaires les plus répandus (et qui nous paraissent les mieux adaptés aux problèmes qui nous concernent) [3].

Les algorithmes génétiques classiques, initialement introduits par Holland [Holland, 1975], s'appuient sur un codage de l'information qu'ils manipulent ainsi que sur un ensemble d'opérateurs génétiques permettant de produire des configurations du problème à partir de configurations parentes. Le codage de l'information, qui doit être défini de manière adaptée pour permettre des recombinaisons génétiques simples et efficaces, peut se faire de multiples façons. Le codage le plus courant utilise des chaînes de bits de longueur fixe représentant la traduction en langage binaire de la valeur de chaque variable (ou gène) d'une solution [Goldberg, 1989]. Un autre codage simple consiste à utiliser directement les valeurs des variables du problème en tant que gènes des individus à manipuler.

L'algorithme 4.1 donne le schéma général d'un algorithme génétique classique. Le processus commence par générer une population de N configurations, puis évalue les individus par une fonction d'adaptation (ou de *fitness*) définie en fonction du problème à optimiser (généralement la valeur d'adaptation est égale au score $f(x)$ de

3. Pour une présentation générale des approches évolutionnaires, le lecteur intéressé pourra se référer à [Bäck *et al.*, 1997].

Algorithme 4.1 : Pseudo-code d'un algorithme génétique classique

Données :
Une fonction $f : \mathcal{S} \to \mathbb{R}$,
La taille N des populations.
Résultat :
Une solution $x^* \in \mathcal{X}$.

```
début
    x* = x ; f* = f(x);
    Initialiser aléatoirement la population P et créer la population vide P';
    tant que le critère d'arrêt n'est pas rencontré faire
        Calcul de la valeur d'adaptation pour tous les individus de P;
        tant que |P'| < N faire
            Sélection dans P en fonction de la valeur d'adaptation;
            Croisements;
            Mutations;
            Insertion des individus produits dans P';
        fin
        P=Sélection de N individus dans P ∪ P';
        P' = ∅ ;
    fin
    si f* > min_{x'∈P} f(x') alors
        x* = argmin_{x'∈P} f(x');
        f* = f(x*);
    fin
fin
```

la fonction à optimiser). Le processus entre ensuite dans une phase de reproduction qui se poursuit tant que l'on n'a pas atteint un nombre suffisant de nouveaux individus. À chaque itération de cette phase de reproduction, trois opérateurs génétiques interviennent :

- Sélection : Choix, dans la population P, des individus à croiser pour produire de nouveaux individus. Cette sélection peut se faire par utilisation d'une probabilité proportionnelle à la valeur d'adaptation des individus (*Roulette Wheel* [Holland, 1975]), par rang (probabilité de sélection proportionnelle au rang de l'individu dans la population selon sa valeur d'adaptation), par tournoi (utilisation d'une probabilité proportionnelle sur des paires d'individus) ou de façon uniforme (même probabilité de sélection pour tous les individus) ;
- Croisement : recombinaisons des gènes de deux individus parents pour former de nouveaux individus (généralement deux). Cette opération de croisement peut se faire de très nombreuses façons qui dépendent du codage du problème. Les croisements les plus répandus sont les croisements en 1, 2 ou k points (tels que représentés par la figure 4.2). Les croisements uniformes, où l'on choisit

la valeur de chaque élément du codage équiprobablement dans l'un ou l'autre des deux parents, sont eux aussi fréquemment employés.

– Mutation : Remplacement d'une ou de plusieurs valeurs du génotype d'un individu par une autre valeur de manière aléatoire (on peut aussi envisager une mutation par utilisation d'une recherche locale). Cet opérateur a pour objectif de diversifier la recherche et d'éviter une convergence prématurée due à la présence trop importante de certains gènes. La probabilité d'application de cet opérateur doit être judicieusement choisie pour permettre une diversification de la population favorisant la convergence de la recherche vers des solutions s'approchant de l'optimum global du problème. ;

Une fois la phase de reproduction terminée, l'algorithme choisit N configurations dans les deux populations (des parents et des enfants) pour former la génération suivante. Le processus se poursuit tant que l'on n'a pas rencontré un critère d'arrêt donné (qui peut correspondre à un nombre de générations maximal ou un nombre de générations sans amélioration de la meilleure solution rencontrée).

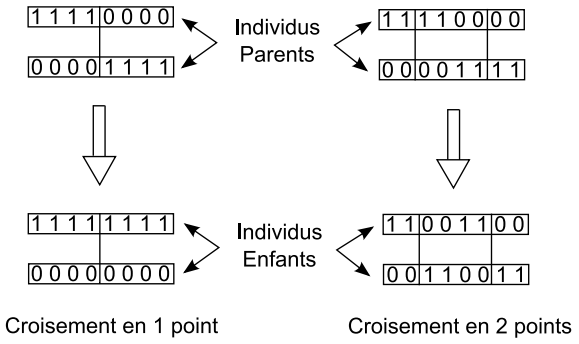

FIGURE 4.2 – Croisements en 1 et 2 points

De très nombreux algorithmes génétiques ont été proposés pour la résolution de problèmes d'optimisation, celui décrit ici (algorithme 4.1) n'en est qu'un exemple très simple. Il est communément admis que pour être réellement efficace, un algorithme doit être adapté au problème considéré. Ainsi, les différents opérateurs peuvent être définis différemment selon les applications. Le fonctionnement général décrit par l'algorithme 4.1 peut lui aussi être remis en question, notamment pour ce qui est de la manière dont on manipule les différentes populations (par exemple le mode de remplacement de la population courante). Un aperçu des algorithmes génétiques et de leur mise en œuvre pour la résolution de problèmes d'optimisation est donné dans [Poli *et al.*, 2008]. Le problème crucial du paramétrage des algorithmes génétiques, notamment pour le choix des probabilités appliquées aux différents opérateurs génétiques, est exploré en détail dans [Lobo *et al.*, 2007].

4.3 Optimisation multi-objectifs

Alors que pour l'ensemble des approches présentées dans les sections précédentes nous nous sommes placés dans un cadre de problèmes à objectif unique, de nombreux problèmes d'optimisation réels impliquent la prise en compte de critères multiples. Par exemple, lors de l'achat d'un produit, il convient généralement de rechercher un compromis entre deux critères contradictoires : on cherche à maximiser la qualité du produit à acheter, tout en minimisant ce que l'on va devoir payer pour l'obtenir. C'est dans ce cadre que se placent les problèmes d'optimisation multi-objectifs. Dans le cas d'un problème de minimisation multi-critères, on a alors :

$$\left|\begin{array}{l} \text{Soit } f : \mathcal{S} \rightarrow \mathbb{R}^m, \\ \text{alors } \hat{\mathcal{S}} \text{ est l'ensemble des minimiseurs de } f \text{ tel que :} \\ \hat{\mathcal{S}} = \Big\{ x \in \mathcal{S} | \forall x' \in \ \mathcal{S}, \Big(\exists i \in \{1,\ldots,m\}, f_i(x) < f_i(x') \Big) \\ \qquad\qquad\qquad\qquad \vee \ \Big(\forall i \in \{1,\ldots,m\}, f_i(x) = f_i(x') \Big) \Big\} \end{array}\right. \tag{4.2}$$

où f_i correspond à la i-ième composante de la fonction f (i-ième critère à optimiser). D'après cette définition, il paraît clair que l'optimum n'est plus une simple valeur tel que cela peut être le cas avec les problèmes mono-objectifs, mais un ensemble de solutions appelé ensemble de compromis ou *front Pareto*. Alors que dans les problèmes mono-objectifs, les relations d'ordre entre les différentes solutions rencontrées sont évidentes (une solution donnée en domine une autre si elle permet d'obtenir un meilleur score de la fonction objectif), la comparaison des éléments dans le cadre des problèmes multi-objectifs est plus délicate. De nombreuses relations de dominance ont été proposées dans la littérature pour distinguer les différentes solutions d'une recherche multi-objectifs, la plus connue et utilisée d'entre elles étant la dominance au sens de *Pareto* [Voorneveld, 2003] que nous présentons maintenant.

4.3.1 La dominance au sens de *Pareto*

Puisque, dans le cadre de problèmes multi-objectifs, les relations d'ordre usuelles ($=$, $<$, $>$, $\leq$, $\geq$) sont difficilement applicables (chaque solution correspondant plus ou moins bien aux différents objectifs du problème), les relations suivantes, dites de relations de dominance au sens de *Pareto*, ont été définies :

Définition 1 - Dominance : *Avec u et v des solutions de l'espace de recherche $\mathcal{S}$ et f une fonction multi-critères à minimiser, on dit que u domine v ($u \prec v$) ssi : $\forall i \in \{1,\ldots,m\}, f_i(u) < f_i(v)$.*

Définition 2 - Faible Dominance : *Avec u et v des solutions de l'espace de recherche $\mathcal{S}$ et f une fonction multi-critères à minimiser, on dit que u domine faiblement v ($u \preceq v$) ssi : $\forall i \in \{1,\ldots,m\}, f_i(u) \leq f_i(v) \ \wedge \ \exists i \in \{1,\ldots,m\}, f_i(u) < f_i(v)$.*

Définition 3 - Non-Dominance : *Avec u et v des solutions de l'espace de recherche $\mathcal{S}$ et f une fonction multi-critères à minimiser, on dit que u est non-dominée par (ou est incomparable avec) v ($u \sim v$) ssi :* $\big(\exists i \in \{1,\ldots,m\}, f_i(u) < f_i(v) \wedge \exists j \in \{1,\ldots,m\}, f_j(v) < f_j(u)\big) \ \vee \ \forall i \in \{1,\ldots,m\}, f_i(u) = f_i(v)$.

Ces relations de dominance permettent de définir deux zones principales par rapport à un point donné x dans l'espace de recherche rapporté à l'espace des objectifs $\mathcal{F}$ (projection de $\mathcal{S}$ dans l'espace des objectifs) : une zone de préférence, qui contient l'ensemble des solutions dominées par x, et une zone de dominance, qui contient l'ensemble des solutions qui dominent x. La figure 4.3 illustre ces différentes zones pour

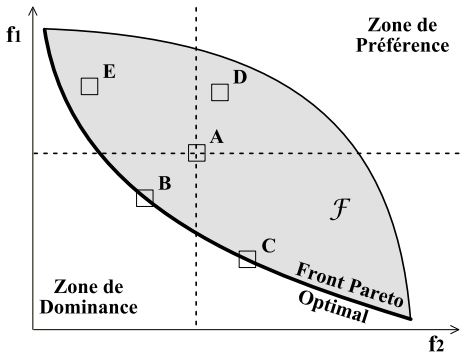

FIGURE 4.3 – Zones de préférence / dominance dans un espace de solutions à deux objectifs

un point A dans un espace à deux objectifs. Alors que le point B appartient à la zone de dominance de A, D est situé dans sa zone de préférence. Les deux autres points, C et E, sont incomparables avec A puisqu'elles le dominent sur un objectif mais lui sont inférieurs sur l'autre. Cette figure fait apparaître également le *front Pareto optimal* du problème, c'est à dire les points qui ne trouvent aucun autre point qui puisse les dominer dans l'espace de recherche (ici, B et C appartiennent à ce *front Pareto optimal*) :

Définition 4 - Front Pareto optimal : *Pour un problème multi-objectifs f donné, le front Pareto optimal $P(f)^*$ est défini par :* $P(f)^* = \{x \in \mathcal{X} | \ \nexists x' \in \mathcal{X}, x' \preceq x\}$.

La résolution de problèmes d'optimisation multi-objectifs consiste approcher au maximum le *front Pareto optimal* du problème. On cherche alors à faire évoluer le front de compromis identifiés (ensemble des solutions non-dominées parmi les solutions examinées) pour le rapprocher de l'ensemble des optima globaux du problème. Pour ce faire, de nombreuses approches, que l'on peut classer en trois catégories, ont été proposées :

– Les approches scalaires, qui transforment le problème multi-objectifs concerné en un problème mono-objectif plus simple à résoudre. Cette catégorie englobe

les méthodes d'agrégation, qui réalisent une somme pondérée des valeurs obtenues par les individus sur chaque critère, les méthodes avec vecteur cible, qui se fixent un but à atteindre dans l'espace des objectifs et cherchent à approcher les solutions de ce point défini, et les méthodes ϵ-contraintes, qui transforment $m-1$ des m objectifs du problème en contraintes ;
- Les approches non scalaire et non *Pareto*, qui traitent séparément chacun des objectifs. On y retrouve notamment les méthodes de sélection parallèle, où les solutions sont évaluées sur chaque objectif indépendamment, les méthodes lexicographique, qui optimisent les objectifs les uns après les autres, et les méthodes avec genres, qui divisent la population d'individus (dans le cadre d'un algorithme génétique) en autant de sous-populations qu'il y a d'objectifs et autorisent uniquement les croisements entre solutions appartenant à des sous-populations différentes ;
- Les approches *Pareto*, qui utilisent les notions de *dominance au sens de Pareto* présentées dans cette section. Ces approches utilisent généralement des algorithmes génétiques pour optimiser simultanément l'ensemble des objectifs du problème.

Plusieurs mesures ont été proposées pour évaluer les fronts de compromis trouvés par les différentes méthodes de résolution. On distingue deux catégories de mesures : les métriques relatives, qui permettent des comparaisons entre fronts, et les métriques absolues, qui permettent de juger un front de compromis de manière isolée. Les mesures les plus répandues sont :

- La métrique d'espacement [Schott, 1995], qui mesure l'uniformité de la répartition des solutions composant la surface de compromis ;
- L'hypervolume [Zitzler, 1999], qui permet une approximation du volume compris sous la courbe formée par les solutions composant la surface de compromis ;
- La métrique $\mathcal{C}$ [Zitzler, 1999], qui permet de comparer deux fronts en calculant la proportion de compromis de l'un des deux fronts à être faiblement dominés par des éléments de l'autre front ;

Ces mesures ont permis d'identifier des tendances quant aux performances des différentes approches de résolution. Il a notamment été établi que, bien qu'elles permettent de simplifier le problème de la résolution multi-objectifs, les méthodes scalaires ont des difficultés à trouver de bonnes solutions pour certains types de problèmes, se révélant bien souvent très sensibles à la forme du front (difficultés par exemple lors de problèmes à front *Pareto* non-convexes). Les méthodes non-scalaire non *Pareto* semblent présenter, quant à elles, l'inconvénient de favoriser bien souvent l'optimisation de certains objectifs au détriment des autres. Nous nous concentrons alors ici sur les dernières approches, les approches *Pareto*, qui ont récemment montré, par l'utilisation de meta-heuristiques tels que les algorithmes génétiques, de très bonnes performances, et ce quelle que soit la forme du front *Pareto* optimal du problème. Plus spécifiquement, nous nous limitons à la présentation d'un algorithme génétique qui a montré ses grandes capacités à produire des surfaces de compromis

de bonne qualité : le *Strength Pareto Evolutionary Algorithm* (*SPEA*) [4].

4.3.2 Le *Strength Pareto Evolutionary Algorithm*

Bien que la population soit censée évoluer au fur et à mesure vers le front *Pareto* optimal du problème, il est peu probable de pouvoir collecter un ensemble de solutions bien réparties sur la totalité de ce front à partir de la seule population finale. Par ailleurs, pour permettre une bonne évolution de la population, il paraît important d'être capable de déterminer, à chaque génération, quelle solution est dominée et quelle autre ne l'est pas au regard des individus précédemment rencontrés au cours de la recherche. Considérant la sauvegarde des meilleurs individus rencontrés au cours de la recherche comme un aspect central de l'optimisation multi-objectifs, l'algorithme *SPEA* introduit par Zitzler [Zitzler, 1999] est l'exemple type d'un algorithme évolutionnaire élitiste [5]. Alors que dans de nombreuses approches, l'élitisme est assuré en sélectionnant un nombre donné de meilleurs individus de la population pour les réinjecter dans la génération suivante, l'algorithme *SPEA* consacre une structure particulière pour sauvegarder ces individus.

Tel que l'illustre la figure 4.4 qui présente son fonctionnement général, l'algorithme *SPEA* (décrit par le pseudo-algorithme 4.2) manipule en fait trois populations d'individus : une archive externe $\bar{P}$, qui collecte les individus non-dominés rencontrés au cours de la recherche (et qu'il faut donc mettre à jour à chaque génération en fonction des nouveaux individus non-dominés et des anciens individus nouvellement dominés), une population courante P_t, qui contient des individus dominés qui se révèlent utiles pour maintenir une certaine diversité et ne pas s'enfermer dans des optima locaux, et une population temporaire, qui contient les individus résultant de la phase de reproduction des individus.

Lorsque la taille l'archive $\bar{P}$ devient trop conséquente, un mécanisme de réduction est mis en œuvre. Ce processus de réduction a deux principaux objectifs :

- Simplifier les opérations de mises à jour des populations : à chaque génération, l'algorithme doit comparer l'ensemble des nouveaux individus avec les individus contenus dans l'archive $\bar{P}$ pour déterminer quels nouveaux individus sont non-dominés et quels individus de l'archive sont dominés par des nouveaux individus. Cette opération peut s'avérer très coûteuse si le nombre d'individus de l'archive devient trop important. Le processus de réduction vise alors à maintenir une taille d'archive raisonnable ;
- Éviter la sur-représentation : lorsque le nombre d'individus situés dans une zone donnée de l'espace des objectifs devient trop important, l'algorithme

4. Pour une présentation plus générale des différentes approches et métriques dédiées aux problèmes multi-objectifs, le lecteur intéressé pourra se référer à [Ehrgott and Gandibleux, 2000], [Collette and Siarry, 2002] ou [Barichard, 2003].

5. L'élitisme dans les algorithmes génétiques correspond à un mécanisme permettant de réintroduire les meilleurs individus d'une génération dans les générations suivantes. Une des premières implémentations de ce mécanisme a été présentée dans [De Jong, 1975]. Permettant bien souvent d'améliorer les performances des algorithmes, de nombreuses approches ont depuis employé ce genre de mécanisme.

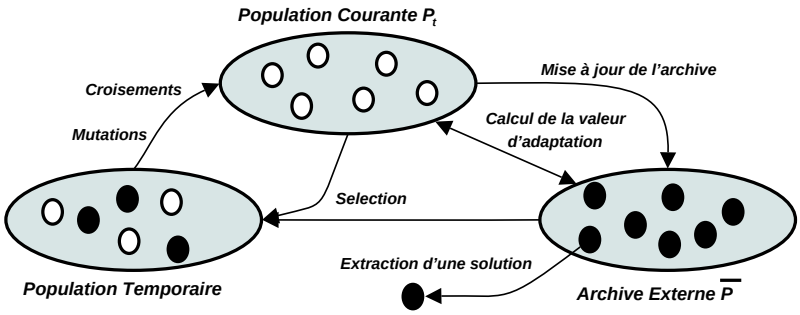

FIGURE 4.4 – Fonctionnement général

Algorithme 4.2 : Pseudo-code de l'algorithme SPEA

Données :
Une fonction $f : \mathcal{S} \rightarrow \mathbb{R}^m$,
Une taille maximale Max pour l'archive $\bar{P}$.
Résultat :
Un ensemble de compromis non-dominés $\bar{P}$.

```
début
    Initialiser la population P_0 et créer l'archive externe vide P̄ = ∅;
    Mise à jour de P̄ avec les individus non dominés de P_0;
    tant que le critère d'arrêt n'est pas rencontré faire
        Calcul de la valeur d'adaptation pour tous les individus de P̄ ∪ P_t;
        Sélection dans P̄ ∪ P_t en fonction de la valeur d'adaptation;
        Croisements;
        Mutations;
        P_t = ∅;
        Insertion dans P̄ des nouveaux individus non-dominés;
        Insertion dans P_t d'individus de P̄ nouvellement dominés;
        Insertion dans P_t de nouveaux individus dominés;
        si |P̄| > Max alors
            Réduction de P̄ par utilisation d'une technique de clustering;
        fin
    fin
fin
```

risque de sélectionner un trop grand nombre de représentants de cette zone pour le processus de reproduction. Il en résulte que les nouveaux individus tendent à appartenir eux aussi à cette zone et que l'algorithme aura alors du mal à à couvrir la totalité du front *Pareto*. Le processus de réduction vise alors à rééquilibrer la représentation des zones de l'espace des objectifs.

Le mécanisme de réduction implémenté dans l'algorithme *SPEA* utilise un processus de clustering des solutions. La mesure de distance entre individus la plus couramment employée semble être une mesure de distance euclidienne appliquée aux scores des individus sur les différents objectifs du problème :

$$\delta(x, x') = \sqrt{\sum_{i=1}^{m} (f_i(x) - f_i(x'))^2} \tag{4.3}$$

La technique de clustering utilisée par *SPEA* s'apparente à une méthode de clustering hiérarchique (voir au chapitre 2) : au début de la procédure, chaque individu constitue son propre groupe, puis on fusionne deux à deux les groupes les plus proches jusqu'à atteindre le nombre de groupes désirés (c'est à dire le nombre d'individus que l'on souhaite obtenir dans l'archive réduite). Une fois les groupes formés, on choisit un individu représentant dans chacun d'entre eux (par exemple le médoïde, l'individu le plus au centre). Seuls les représentants des groupes sont alors conservés, les autres individus étant tout simplement supprimés.

Alors que dans de nombreux algorithmes génétiques, la valeur d'adaptation des individus pour leur sélection correspond directement au score qu'ils obtiennent selon la fonction objectif du problème, l'algorithme *SPEA* définit un calcul spécifique qui garantit un certain niveau de diversité dans la population. De nombreux mécanismes ont été proposés dans ce sens, notamment les techniques de *sharing*, de *réinitialisation* et de *crowding* [Barichard, 2003]. L'algorithme *SPEA* emploie, quant à lui, sa propre technique qui consiste à utiliser la notion de dominance pour orienter la recherche vers des zones de l'espace des objectifs peu explorées. Ainsi, l'algorithme commence par associer une valeur de "dureté" $S(x)$ à chaque élément x de l'archive externe $\bar{P}$. Cette valeur est calculée selon le nombre d'individus de P_t que l'individu concerné domine faiblement :

$$S(x) = \frac{|\{x' | x' \in P_t \wedge x \preceq x'\}|}{|P_t| + 1} \tag{4.4}$$

La valeur d'adaptation $F(x)$ d'un individu $x \in \bar{P}$ correspond à la valeur de dureté qui lui a été associée : $F(x) = S(x)$. La valeur d'adaptation $F(x')$ d'un individu $x' \in P_t$ est calculée en réalisant la somme des valeurs de dureté de tous les individus de l'archive externe qui le dominent faiblement :

$$F(x') = 1 + \sum_{x \in \bar{P} \ \wedge \ x \preceq x'} S(x) \tag{4.5}$$

La figure 4.5 est un exemple de calcul de valeurs d'adaptation pour les individus de $P_t \cup \bar{P}$ à un moment t donné. Elle donne les valeurs obtenues pour les différents individus dominés ou non-dominés. On observe que le calcul réalisé favorise les

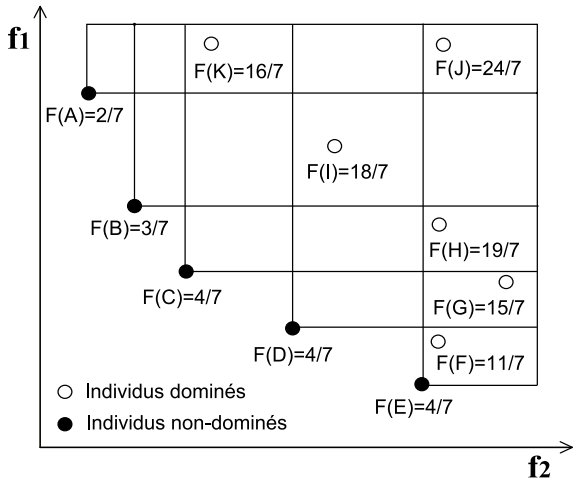

FIGURE 4.5 – Valeur d'adaptation dans SPEA

éléments non-dominés qui ne dominent que peu d'individus [6] : par exemple, l'individu A, qui ne domine que 2 individus, obtient une valeur d'adaptation bien meilleure que l'individu C qui en domine 4. Lorsque l'on s'intéresse aux individus de P_t, on note également que le calcul réalisé favorise les individus dominés par des configurations de $\bar{P}$ qui ne dominent que peu d'individus de P_t : par exemple, la comparaison entre les individus K et H, qui sont pourtant dominés par autant d'individus, penche en faveur de K qui est dominé par A, B et C, des individus de $\bar{P}$ dominants moins d'individus de P_t que les dominants de H. Ce calcul de la valeur d'adaptation oriente donc la recherche vers les zones les moins représentées de l'espace des objectifs et permet alors d'obtenir un ensemble de compromis probablement mieux répartis sur le front *Pareto* que si l'algorithme avait utilisé une valeur d'adaptation classique dépendant directement de la qualité des individus au regard des différents objectifs.

4.4 Application à la recherche d'information

Ayant achevé notre survol des principes fondamentaux de la résolution des problèmes d'optimisation combinatoire, nous nous intéressons maintenant à l'application des techniques d'optimisation, et plus particulièrement des algorithmes génétiques [7], dans le domaine de la recherche d'information. La capacité singulière des algorithmes génétiques à explorer de très grands espaces de recherche les rend

6. Il est à noter qu'avec ce calcul, plus la valeur d'adaptation est petite, plus l'individu a des chances d'être sélectionné pour participer à la phase de reproduction.

7. Bien que quelques travaux aient utilisé d'autres techniques d'optimisation pour la recherche d'information, notamment les colonies de fourmis ou la recherche Tabou. Les algorithmes génétiques sont néanmoins les plus répandus dans le domaine de la recherche d'information. Le lecteur intéressé par l'application des autres meta-heuristiques dans ce domaine pourra se référer à [Picarougne, 2004].

particulièrement adaptés à la recherche d'information où justement l'objectif est de retrouver des solutions dispersées dans les gigantesques ensembles de données que représentent parfois les bases de textes interrogées. D'une manière générale, les applications des algorithmes génétiques en recherche d'information ont porté majoritairement sur :

- La représentation des documents et des requêtes : de nombreux travaux, dont Gordon [Gordon, 1988] est l'un des pionniers, ont porté sur l'utilisation d'algorithmes génétiques pour rechercher des modes de représentation permettant de rendre compte au mieux des concepts abordés par les documents ou les requêtes. Généralement dans ces approches, les algorithmes génétiques sont employés pour altérer la représentation des documents selon des jugements de pertinence de l'utilisateur et adopter une représentation qui permette d'identifier efficacement les documents pertinents ;
- La reformulation de requêtes : les algorithmes génétiques ont été maintes fois utilisés pour la reformulation de la requête exprimée par l'utilisateur, se basant sur des appréciations de pertinence pour transformer les poids des termes qu'elle contient (c'est à dire une expansion de requête par réinjection de pertinence, voir section 1.4.2). La première application des algorithmes génétiques pour cette tâche semble avoir été proposée par Yang dans [Yang, 1993] ;
- Les mesures d'estimation de pertinence : quelques travaux ont concerné l'utilisation d'algorithmes génétiques pour tenter d'améliorer l'efficacité des mesures d'estimation de pertinence utilisées pour retourner des documents en réponse à une requête utilisateur. Notamment, Pathak et al. [Pathak *et al.*, 2000] ont proposé de combiner plusieurs mesures de similarité différentes et de se servir d'algorithmes génétiques pour les pondérer de manière optimisée selon une mesure d'efficacité du système ;
- Le clustering de documents : déjà très présents dans les applications de clustering de données en général, l'utilisation d'algorithmes génétiques pour la catégorisation de documents semble évidente. De nombreuses propositions ont été faites pour permettre la production des groupes de documents les plus utiles à la localisation d'informations pertinentes, en employant notamment les algorithmes génétiques pour prendre en compte des appréciations de l'utilisateur lors de la formation des catégories [Raghavan and Agarwal, 1987] ;

Les algorithmes génétiques semblent alors avoir été appliqués à la plupart des problèmes de la recherche d'information... Ce qu'il ressort néanmoins de l'étude des différentes propositions d'utilisation des algorithmes génétiques en recherche d'information est que ces meta-heuristiques ont surtout été employées comme vecteurs de prise en compte des différentes interactions avec l'utilisateur [López-Pujalte *et al.*, 2003]. Dans ce contexte, de nombreux travaux ont proposé des mécanismes et opérateurs spécifiques à ces applications [Horng and Yeh, 2000; Boughanem *et al.*, 2002]. L'opérateur de croisement semble avoir été tout particulièrement étudié, ses versions classiques ne paraissant pas toujours conduire à

l'obtention des meilleurs résultats (un opérateur tel que le croisement dissocié, par exemple, tend à améliorer significativement l'efficacité de la recherche [Vrajitoru, 1998]).

Cependant, cette application des algorithmes génétiques ne cadre que très peu avec le travail réalisé dans cette thèse (elle peut néanmoins être considérée pour une mise en place future d'interactions avec l'utilisateur). La plupart des approches proposées utilisent en effet plutôt les algorithmes génétiques pour leurs capacités d'apprentissage que comme des meta-heuristiques de résolution de problèmes d'optimisation. Contrairement à l'utilisation que l'on souhaite faire des algorithmes génétiques (notre objectif principal étant de déterminer le sous-ensemble de textes, documents ou fragments de documents, permettant la meilleure description du sujet, voir chapitre 9), les gènes des individus correspondent bien souvent aux différents termes du corpus et l'exploration de l'espace de recherche vise à déterminer quel sous-ensemble de termes peut permettre la meilleure représentation des documents selon la requête de l'utilisateur et les retours de pertinence considérés. Par ailleurs, les techniques de résolution de problèmes d'optimisation multi-objectifs semblent très peu représentées en recherche d'information (excepté quelques travaux isolés tels que l'optimisation multi-objectifs réalisée dans [Fisher *et al.*, 2003] pour considérer simultanément différentes mesures de similarités entre documents et requêtes). Certainement, les utilisations des algorithmes génétiques qui se rapprochent le plus des approches que nous proposons dans cette thèse sont celles données dans [Maulik and Bandyopadhyay, 2000] et [Handl and Knowles, 2007], mais le domaine d'application est la mise en place de techniques de clustering générales (non spécifiques à la catégorisation de documents), ou ceux relatifs à la recherche des phrases les plus représentatives des informations abordées par un document (résumé automatique) [Silla *et al.*, 2004] ou un ensemble de documents (résumé multi-documents) [Liu *et al.*, 2006] en terme d'informativité ou de couverture des sujets abordés, mais ces approches ne sont pas appliquées à la recherche d'information, cherchant à synthétiser l'ensemble des informations sans prendre en compte les besoins informationnels de l'utilisateur. Notons enfin que les algorithmes génétiques ont été employés avec succès pour tirer partie des connexions (*hyperliens*) entre pages *Web* afin de déterminer les pages les plus reliées à un sujet donné (le nombre de connexions entre pages induisant un espace de recherche qui implique l'utilisation de techniques d'optimisation) [Picarougne, 2004].

4.5 Conclusion

Dans ce chapitre, nous avons présenté les bases de l'optimisation combinatoire. Après avoir énoncé différentes méthodes proposées pour approcher les solutions optimales de problèmes comprenant un objectif unique, en nous concentrant notamment sur ce type particulier de meta-heuristiques que constituent les algorithmes génétiques, nous nous sommes penchés sur l'épineux problème de l'optimisation

multi-objectifs. Si la résolution de problèmes mono-objectifs pose déjà de nombreuses difficultés, l'optimisation de problèmes multi-objectifs, qui trouve de très nombreuses applications, représente un challenge important qui implique la définition de notions et techniques spécifiques. Nous avons alors proposé un rapide survol des orientations prises dans ce domaine, pour terminer sur la présentation d'un algorithme génétique dédié à l'optimisation multi-objectifs, le *Strength Pareto Evolutionary Algorithm*, que nous employons à plusieurs reprises dans la suite de cette thèse. Enfin, l'étude des applications des techniques d'optimisation, et plus particulièrement des algorithmes génétiques, en recherche d'information, nous a permis de nous rendre compte du peu de travaux considérant la recherche d'information, du moins la recherche d'information statique (c'est à dire sans intervention de l'utilisateur), comme un problème d'optimisation, qui plus est comme un problème d'optimisation multi-objectifs. Néanmoins, lorsque l'on considère les multiples connections existant entre les documents d'un corpus et les différents facteurs entrant en jeu dans le processus complexe de la recherche d'information, la composition d'un ensemble de textes répondant aux attentes d'un utilisateur constitue un problème qui peut requérir l'utilisation de techniques d'optimisation performantes. C'est dans cette optique d'optimisation des informations apportées à l'utilisateur que nous nous plaçons dans le cadre de cette thèse.

Deuxième partie

Extraction des thématiques

L'objectif central de cette thèse étant de proposer une méthodologie de sélection et de combinaison des fragments de texte répondant au mieux aux besoins en information d'un utilisateur, la première étape du travail consiste naturellement à définir un mode de découpage efficace des documents du corpus considéré. Cette partie se pose alors en introduction à la conception de document à proprement parler, englobant une série de travaux préliminaires dont le but est de déterminer le meilleur moyen d'extraire les passages des textes avec lesquels nous pourrons composer par la suite. Une segmentation thématique des documents nous apparaissant comme le mode de découpage le plus adapté à nos besoins, nous nous focalisons dans un premier temps sur les approches permettant un tel découpage des textes. Cette étude achevée, nous proposons une série d'expérimentations permettant de mettre en évidence les bénéfices potentiels résultant de la prise en compte individuelle des différentes thématiques des documents, tant pour la production d'une liste ordonnée de résultats, que pour la formation de catégories de documents permettant une localisation facilitée des informations recherchées. D'une manière générale, cette partie s'intéresse à la fragmentation des documents et à ses impacts sur les performances des systèmes de recherche d'information. Elle a pour principal objectif de définir les données manipulées lors de la composition finale de document, qui constitue l'aboutissement de ce travail de thèse.

Chapitre 5

Segmentation thématique

Ce chapitre porte sur les méthodes de segmentation thématique, dont le but est d'identifier les principales ruptures discursives des documents pour en extraire des fragments de texte thématiquement homogènes. Après avoir étudié les approches proposées dans la littérature pour effectuer un tel découpage des documents, nous présentons les méthodes et mesures que nous avons mises au point pour dépasser certaines limites identifiées.

Sommaire

5.1 La Segmentation thématique de textes 104
5.2 Méthodes de segmentation thématique 105
5.3 Méthodologies d'évaluation 108
5.3.1 Constitution d'une segmentation de référence 108
5.3.2 Mesures d'évaluation 110
5.4 Vers une segmentation globale et cohérente des textes 112
5.4.1 ClassStruggle : mise en concurrence de groupes thématiques . . . 115
5.4.2 SegGen : optimisation multi-objectifs des segments 122
5.4.3 Évaluation des systèmes 127
5.5 Vers un mode d'évaluation plus équitable 133
5.5.1 Analyse de la mesure d'évaluation WindowDiff 134
5.5.2 Prise en compte des risques encourus 139
5.5.3 Respect des différences 142
5.6 Conclusion 147

5.1 La Segmentation thématique de textes

Considérant un document comme la trace écrite d'un discours organisé [1], de nombreux chercheurs ont, à maintes reprises, insisté sur l'apport informationnel que pouvait offrir une prise en compte de la structuration interne des textes [Jacques and Rebeyrolle, 2006] [2]. Bien qu'un document soit supposé traiter d'une même thématique [3] générale tout au long de son discours, différentes étapes lui sont parfois nécessaires pour atteindre son objectif principal (c'est à dire aborder l'idée majeure pour laquelle il a été construit). Ces différentes étapes peuvent représenter des passages aux thématiques relativement déconnectées, tournant autour d'un même sujet général mais en abordant différents aspects. Par exemple, la thématique générale de cette thèse est la recherche d'information, et l'objectif final la présentation d'un système de composition de documents, mais les différents chapitres en décrivent tous un aspect différent, certains posant les bases du discours, d'autres présentant des travaux préliminaires, etc... Dans de tels documents, l'existence d'une structuration physique du texte (découpage en parties, chapitres, sections, sous-sections, paragraphes, etc...) facilite la prise en compte de l'organisation du discours. Mais de nombreux documents ne présentent ni structuration ni formatage d'aucune sorte [4]. On peut alors recourir aux méthodes de segmentation thématique dont l'objectif est d'identifier les ruptures les plus importantes d'un texte afin de le découper en passages thématiquement homogènes. Salton définit ces passages, appelés ci-après segments thématiques, comme des "extraits de texte possédant de forts liens thématiques internes et étant en grande partie déconnectés des extraits adjacents" [Salton *et al.*, 1996]. Ainsi, un segment thématique n'est défini que par opposition aux autres segments qui l'entourent : "La notion de 'thématique' est manifestement une manière intuitive de décrire le principe unificateur selon lequel une partie d'un texte relève d'un sujet donné et la partie suivante d'un autre sujet bien différent [...] plutôt que de chercher à définir ce qu'est une 'thématique', nous devrions nous concentrer sur la description de ce que nous appelons une 'transition entre thématiques' " [Brown and Yule, 1983]. Selon la méthode, ces transitions sont déterminées selon différents marqueurs : marqueurs linguistiques [Chafe, 1979], marqueurs de cohésion lexicale [Hearst, 1997], marqueurs de cohésion sémantique [Morris and Hirst, 1991],

1. Pour une étude approfondie de la notion de document, se référer à [Pédauque, 2003].

2. Schlieder et Meuss affirment même qu'"ignorer la structure du document revient à ignorer sa sémantique" [Schlieder and Meuss, 2002].

3. En linguistique, on définit généralement le thème comme "l'élément d'un énoncé qui est réputé connu par les participants à la communication". Il est souvent opposé au rhème, qui est l'information nouvelle apportée par l'énoncé. Ces deux notions sont souvent interprétées de la manière suivante : le thème correspond à ce dont on parle, le rhème représente ce que l'on en dit.

4. Par ailleurs, il n'est pas rare que la structuration physique présentée ne corresponde pas à la structuration réelle du discours. Par exemple, il se peut que tel ou tel changement de paragraphe dans un document ait uniquement été opéré pour de simples raisons de présentation [Callan, 1994]. Il a été établi qu'une segmentation des textes en fonction des seules marques typographiques est rarement suffisante [Al-hawamdeh and Willett, 1989].

etc... Alternativement, la segmentation thématique des textes peut être considérée comme le résultat d'un processus de regroupement d'unités du discours telles que des phrases, des mots ou des paragraphes. Pour Kozima [Kozima, 1993], un segment thématique correspond à une "suite de propositions ou de phrases faisant preuve d'une forte cohésion locale". Le degré de granularité de la segmentation dépend alors de la taille des unités à regrouper [5].

Une autre question concerne alors les relations entre segments à produire : doivent-ils être organisés de manière linéaire (un segment en suit un autre) ou bien plutôt être organisés de manière hiérarchique (voire des organisations plus complexes [Mann and Thompson, 1988]), de façon à faire apparaître différents niveaux de structuration ? Selon la définition donnée à la notion de thématiques d'un document, une structure hiérarchique pourrait paraître appropriée, puisque l'on peut toujours englober "ce dont on parle" dans un cadre de discours plus général. De nombreuses théories du discours, telles que celle donnée dans [Grosz and Sidner, 1986], suggèrent une telle segmentation hiérarchique des documents. Néanmoins, ces modèles de segmentation sont généralement plutôt utilisés pour une segmentation très fine du discours [Moore and Pollack, 1992], la complexité de la tâche les rendant difficilement exploitable [Hearst, 1994; Yaari, 1997]. Par ailleurs, il est à noter qu'une segmentation hiérarchique peut être obtenue par applications successives d'une méthode de segmentation linéaire sur les différents segments découverts. En effet, la sensibilité de la plupart des méthodes, notamment les méthodes travaillant par étude des variations de cohésion lexicale, dépend du texte à segmenter : lorsque les variations de cohésion sont plus faibles, la méthode réalise une segmentation plus "fine" du texte. Hearst fait par ailleurs remarquer que, lorsque l'objectif est de détecter les principales transitions thématiques d'un texte pour en découvrir la structure, une segmentation linéaire est suffisante [Hearst, 1997].

Ainsi, la tâche de segmentation thématique revient à diviser un document en secteurs contigus par la détermination de frontières entre unités du texte :

Définition 5 - Segmentation : *Étant donné un texte représenté par une séquence de n unités $\mathcal{U} = \{u_1, \ldots, u_n\}$ (phrases ou paragraphes généralement), une segmentation est définie par un couple $(\mathcal{U}, \mathcal{B})$ où $\mathcal{B}$ est un ensemble de m (avec $m \leq n-1$) frontières distinctes $\mathcal{B} = \{b_1, \ldots, b_m\}$ tel que $\forall i \in \{1, \ldots, m\}, 1 \leq b_i < n$. Une segmentation $\mathcal{S} = (\mathcal{U}, \mathcal{B})$ détermine alors un ensemble de $m+1$ parties de texte $u_1 \ldots u_{b_1} \mid \ldots \mid u_{b_{m-1}+1} \ldots u_{b_m} \mid u_{b_m+1} \ldots u_n$.*

5.2 Méthodes de segmentation thématique

Étant donné le nombre d'applications (alignement de textes, résumé automatique, extraction d'information, recherche documentaire, etc...) que peut avoir la

5. Pour nos expérimentations (sections 5.4.3 et 5.5), l'unité de discours choisie est la phrase. D'autres unités telles que le mot ou le paragraphe (à l'instar de [Hearst, 1997]), auraient tout aussi bien pu être considérées.

segmentation thématique des textes, de très nombreuses méthodes ont été proposées. Ces méthodes peuvent se classer en trois catégories :

- Les méthodes linguistiques, qui segmentent les textes en fonction de l'apparition de termes connus pour être des marqueurs d'introduction ou de conclusion de thématiques [Grosz and Sidner, 1986]. À l'inverse, des marqueurs linguistiques de continuité thématique peuvent aussi être considérés [Passonneau, 1993] ;
- Les méthodes probabilistes, qui réalisent un apprentissage sur des textes structurés pour définir des probabilités de changement thématique. Parmi les méthodes appartenant à cette catégorie, citons celles présentées dans [Beeferman *et al.*, 1997] et [Bigi *et al.*, 1998], qui définissent un modèle de langue pour déterminer les frontières thématiques les plus probables, celle proposée dans [Amini *et al.*, 2000], qui se fonde sur un modèle probabiliste basé sur la détermination de chaînes de Markov cachées, ou celle donnée dans [Caillet *et al.*, 2004], qui utilise un processus d'apprentissage pour produire un clustering des termes les plus proches et ainsi déterminer les plus importantes ruptures thématiques d'un texte ;
- Les méthodes statistiques, qui s'appuient majoritairement sur la distribution des mots dans le texte pour en déterminer les changements thématiques, la répétition des termes pouvant s'avérer être un bon indicateur de cohésion [Hearst, 1994]. L'une des premières méthodes de ce type est celle présentée dans [Skorochod'ko, 1971].

Du fait de leurs performances et de leur simplicité, les méthodes statistiques sont certainement les plus répandues. Certaines méthodes des deux autres catégories obtiennent elles aussi de bons résultats mais leur portabilité peut s'avérer difficile [Utiyama and Isahara, 2001]. Nous nous limiterons ici à la présentation des méthodes de segmentation statistiques.

La plupart des méthodes statistiques font l'hypothèse que le degré de cohésion d'un texte peut être estimé en considérant le taux de répétition des termes qu'il contient [Halliday and Hasan, 1976; Walker, 1992] : un nombre important d'occurrences des mêmes termes dans une zone donnée dénote d'une forte cohésion des unités dans la zone concernée. Les méthodes tendent alors à étudier les variations de cet inventaire lexical au fil du texte, pour déterminer des frontières dans les zones possédant les plus faibles volumes de termes répétés.

Pour identifier ces zones de faible cohésion, certaines méthodes (telles que celles présentées dans [Utiyama and Isahara, 2001] ou [Galley *et al.*, 2003] par exemple) établissent des *chaînes lexicales* entre les occurrences d'un même terme [6] (voir la figure 5.1). Lorsqu'un terme se répète dans un intervalle plus ou moins court (ap-

6. Et éventuellement entre des synonymes, des hyperonymes, des hyponymes ou des termes statistiquement liés (co-occurrences des termes) [Morris and Hirst, 1991].

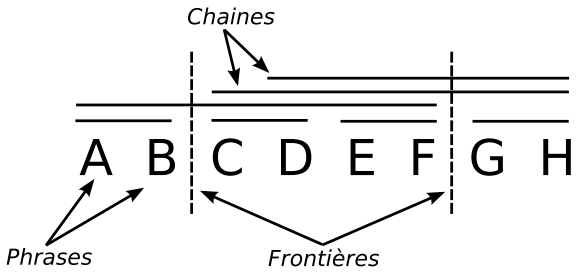

FIGURE 5.1 – Exemple de chaînes lexicales

pelé hiatus[7]), une chaîne lexicale est créée entre ses occurrences. Des frontières thématiques sont alors déterminées aux endroits où le nombre de chaînes est minimal, supposant qu'un faible nombre de chaînes traduit une faible cohésion du texte.

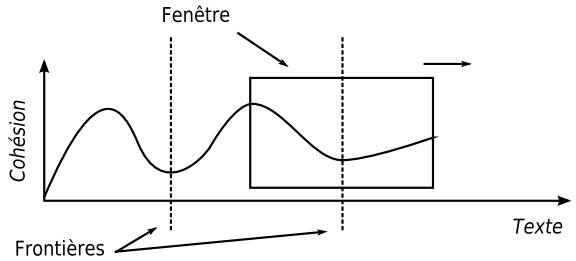

FIGURE 5.2 – Exemple de fenêtre de calcul

D'autres méthodes déterminent une *fenêtre de calcul* (*Sliding Window*) qui parcourt le texte pour calculer ses variations de cohésion (voir figure 5.2) et déterminer des frontières aux endroits où celle-ci est la plus faible [Hearst, 1994].

Réalisant une analogie entre segmentation et recherche d'information, de nombreuses approches tentent d'utiliser les modèles et mesures établis dans ce domaine pour estimer la cohésion des zones de texte de manière plus efficace que par la seule considération des répétitions de termes [Salton *et al.*, 1996]. Ainsi, plutôt que de considérer les termes individuellement, il est possible de les inclure dans une entité, c'est à dire l'unité de discours utilisée (phrase, paragraphe, etc...), afin de les replacer dans le contexte où ils ont été employés. L'utilisation d'une mesure comme la mesure *Cosine* du modèle vectoriel (section 1.3.2) par exemple permet de considérer l'ensemble des termes des unités à comparer. Ce que l'on mesure alors n'est plus uniquement la présence de termes communs mais aussi le degré dans lequel les unités

7. Notons que, plutôt que de déterminer un hiatus de manière empirique, ce qui peut s'avérer difficile, [Sitbon and Bellot, 2005] proposent de créer des chaines lexicales entre toutes les répétitions de termes du texte et d'utiliser une pondération qui dépend, pour chaque chaîne et chaque position du texte, de la distance à l'occurrence la plus proche du terme correspondant à la chaîne concernée.

comparées n'emploient pas des mots différents[8].

Enfin, notons que, de la même manière que pour les systèmes de recherche d'information, les performances des systèmes de segmentation dépendent beaucoup de la nature des documents considérés. Par exemple, les méthodes de segmentation auront plus de facilités à segmenter efficacement des documents techniques, où le vocabulaire reste relativement limité, que des textes narratifs, où le champ lexical est plus vaste [Hearst, 1997]. Par ailleurs, notons que les méthodes de segmentation rencontrent souvent plus de difficultés à segmenter les textes écrits en français que des textes en langue anglaise [Bestgen and Piérard, 2006]. Ceci peut s'expliquer par le fait que la langue française supporte moins bien les répétitions que la langue anglaise et que donc, les méthodes disposent de moins d'indices de cohésion. Afin d'augmenter le nombre de liaisons entre termes, et améliorer ainsi les capacités des méthodes à estimer la cohésion thématique des différentes zones d'un texte, certaines approches réalisent une étude statistique de co-occurrences de termes [Morris and Hirst, 1991; Ferret, 2002], c'est à dire qu'elles cherchent à apprendre quels mots sont souvent associés à quels autres à partir d'un corpus de textes d'apprentissage. Cela permet de considérer, en plus des répétitions de termes, des associations entre termes probablement proches puisqu'utilisés conjointement dans de nombreux textes.

5.3 Méthodologies d'évaluation

L'évaluation des méthodes de segmentation se fait généralement par comparaison des segmentations produites avec des segmentations de référence. Cette section commence par explorer les possibilités de constitution de ces segmentations de référence, puis présente quelques mesures d'évaluation couramment employées.

5.3.1 Constitution d'une segmentation de référence

De la même manière qu'il est difficile de produire des ressources sur lesquelles évaluer les systèmes de recherche d'information, la constitution d'un corpus de textes annotés pour contenir des segmentations de référence auxquelles comparer les segmentations produites par les méthodes peut s'avérer une tâche particulièrement difficile. La première piste qui vient à l'esprit lorsque l'on cherche un moyen de constituer un corpus de segmentations de référence est certainement la considération de jugements humains, tel que c'est le cas pour la recherche d'information dans la campagne *TREC* par exemple (voir section 3.2). Néanmoins cela pose de nombreux problèmes. Tout d'abord, les ressources nécessaires pour la réalisation d'une telle tâche (nombre de personnes impliquées, temps passé par chaque expert, etc...) risquent de s'avérer très contraignantes. En effet, il ne s'agit pas ici de faire un jugement binaire sur la pertinence ou non d'un document par rapport à une requête, ce qui est déjà

8. À l'inverse des répétitions de termes, l'apparition de mots nouveaux dans une unité (par rapport à l'unité qui la précède) peut en effet être un indicateur de changement thématique [Language, 1991].

très coûteux mais n'implique pas forcément de lire l'intégralité du document pour prendre une décision. La segmentation manuelle d'un texte implique d'en saisir l'organisation interne, ce qui induit un volume d'efforts considérable. Si l'on souhaite disposer de corpus suffisamment grands pour être significatifs, l'annotation manuelle de corpus pour l'évaluation de méthodes de segmentation se révèle être une tâche colossale. Mais ce n'est pas le principal problème ! L'aspect subjectif de la segmentation d'un texte est terriblement problématique : il est très rare que les annotateurs s'accordent sur une segmentation donnée [Hearst, 1997]. En effet, les différences de sensibilité aux changements thématiques ou les différences de point de vue sont telles qu'il est difficile de trouver un consensus entre les annotateurs. Des propositions de constitution de segmentations de référence ont été faites (voir par exemple [Nakatani *et al.*, 1995]), mais les corpus disponibles à ce jour sont bien loin d'être suffisamment grands pour réaliser une évaluation qui permette de faire apparaître des tendances significatives.

Une autre piste pour la constitution de segmentations de référence pourrait être d'utiliser des textes pré-formatés : les paragraphes existent déjà dans la version originale du document et les méthodes peuvent être évaluées sur leur capacité à les repérer. Néanmoins, du fait de la présence fréquente de phrases de transition dans les documents structurés, les méthodes risquent d'avoir beaucoup de difficultés à retrouver les différents paragraphes et l'évaluation risque alors de n'être pas très probante. Par ailleurs, tel qu'énoncé plus haut, la structure physique d'un texte ne correspond pas toujours à sa structure thématique.

En fait, pour obtenir des résultats significatifs, il faut que les transitions thématiques soient suffisamment marquées pour pouvoir différencier les méthodes sans ambiguïté. Si les transitions sont floues, il est en effet difficile de comparer les méthodes puisque nous mêmes, en tant qu'humains, risquons d'avoir les mêmes désaccords. Par ailleurs, il est difficile de tirer quelque conclusion que ce soit lorsqu'aucune méthode ne produit ce que l'on attend. Par conséquent, la constitution de corpus d'évaluation se fait généralement par concaténation d'articles [Choi, 2000] : un certain nombre de textes sont mis bout à bout et les méthodes de segmentation sont évaluées sur leur capacité à retrouver les frontières entre articles. Des expérimentations sur ce genre de textes peuvent paraître ne pas représenter la réalité, puisque les documents que l'on segmente sont bien moins homogènes que des documents réels, mais ces transitions franches permettent d'observer des tendances claires. De plus, au vu des résultats obtenus par les méthodes actuelles, le niveau de difficulté induit par ce type de corpus paraît bien suffisant.

Dans la suite de ce chapitre, nous noterons $\mathcal{R} = (\mathcal{U}, \mathcal{B}_\mathcal{R})$ la segmentation de référence ainsi obtenue sur le document $\mathcal{U}$ (composé de plusieurs articles donc) et $\mathcal{H} = (\mathcal{U}, \mathcal{B}_\mathcal{H})$ celle produite sur ce même document par la méthode à évaluer. Par ailleurs, nous noterons $nom\_corpus(n, m)$ les différents corpus constitués, *nom_corpus* correspondant au nom du corpus, n correspondant au nombre moyen de phrases dans les documents de ce corpus et m représentant le nombre d'articles

ayant été concaténés pour former chacun des textes de ce corpus. Par exemple, un corpus noté $AP(100, 4)$ contient des textes d'une taille moyenne de 100 phrases et résultant de la concaténation de quatre articles du corpus AP (présenté en section 3.2).

5.3.2 Mesures d'évaluation

Parmi les mesures d'évaluation actuelles des méthodes de segmentation, on retrouve couramment les mesures classiques de *Rappel* et de *Precision* empruntées à la recherche d'information. Leurs définitions sont ici différentes puisque l'on ne compare pas un ensemble de documents retournés avec un ensemble de documents pertinents mais un ensemble de frontières produites $\mathcal{B}_\mathcal{H}$ avec un ensemble de frontières de référence $\mathcal{B}_\mathcal{R}$:

$$Rappel(\mathcal{H}, \mathcal{R}) = \frac{|\mathcal{B}_\mathcal{R} \cap \mathcal{B}_\mathcal{H}|}{|\mathcal{B}_\mathcal{R}|} \tag{5.1}$$

$$Precision(\mathcal{H}, \mathcal{R}) = \frac{|\mathcal{B}_\mathcal{R} \cap \mathcal{B}_\mathcal{H}|}{|\mathcal{B}_\mathcal{H}|} \tag{5.2}$$

Le rappel correspond alors au ratio de frontières de référence à avoir été identifiées par la méthode à évaluer et la précision représente le ratio de frontières produites à appartenir à la segmentation de référence. Bien que fréquemment employées, ces deux mesures présentent deux limites majeures [Beeferman *et al.*, 1997] :

- À l'instar des observations réalisées pour la recherche d'information (section 3.3.1), ces deux mesures sont en corrélation inverse : alors que le rappel a tendance à augmenter lorsque le nombre de frontières déterminées par la méthode augmente, la précision a tendance à diminuer dans le même temps. Si ce phénomène peut poser un problème d'interprétation lors de leur utilisation pour l'évaluation de systèmes de recherche d'information, les difficultés qu'il induit sont ici bien plus importantes : alors que lors de l'évaluation d'une liste de documents, il est possible de déterminer un nombre donné de documents à considérer, le nombre de frontières de la segmentation à évaluer n'est pas paramétrable *a posteriori* (et n'est d'ailleurs bien souvent pas réglable *a priori* non plus, la plupart des méthodes ne permettant pas de fixer le nombre de frontières à déterminer). La comparaison entre deux méthodes de segmentation déterminant des nombres différents de frontières est alors difficile ;
- Ces critères ne prennent en compte que les correspondances exactes entre la segmentation de référence et celle à évaluer. Un décalage d'une frontière d'une phrase ou deux est alors aussi pénalisant que l'absence de frontière dans cette zone de texte. Il semble néanmoins naturel de considérer ce type d'erreur mineure avec moins de sévérité.

Dans le but de dépasser ces limites, Beeferman et al. ont proposé une mesure d'évaluation alternative, la *Pk-mesure* [Beeferman *et al.*, 1997], qui correspond à

un degré de désaccord entre les deux segmentations (i.e., $\mathcal{H}$ and $\mathcal{R}$) pour ce qui est de l'appartenance de phrases à un même segment. Plus précisément, la mesure passe en revue tous les couples de phrases u_i et u_j séparées l'une de l'autre par une distance de k phrases (on a alors $|i - j| = k$) pour déterminer si les deux phrases considérées appartiennent à un même segment ou non. Si, pour chacune de ces paires de phrases, la réponse à cette question n'est pas la même selon que l'on considère l'une ou l'autre des deux segmentations, la mesure incrémente un compteur de désaccords rencontrés. Le résultat final correspond à ce nombre de désaccords normalisé par le nombre de paires de phrases considérées. Cette mesure permet de régler les deux problèmes évoqués ci-dessus, conduisant à l'obtention d'un score unique et considérant différemment les erreurs selon leur importance.

Après avoir procédé à une analyse de cette mesure, Pevzner et Hearst ont néanmoins identifié un certain nombre de biais [Pevzner and Hearst, 2002] :

- L'absence d'une frontière à une position donnée (frontière manquante) est plus pénalisée que l'insertion d'une frontière n'existant pas dans la référence (frontière abusive) ;
- La mesure ne prend pas en compte toutes les erreurs, certaines pouvant être cachées par d'autres ;
- Les décalages de frontières sont trop pénalisés ;
- La mesure est sensible aux variations de taille des segments.

Par conséquent, Pevzner et Hearst ont proposé une nouvelle mesure d'évaluation qui s'inspire de la *Pk-mesure*, la mesure $WindowDiff$, qui considère le nombre de frontières entre deux phrases séparées par une distance k [Pevzner and Hearst, 2002] :

$$\mathit{WindowDiff}(\mathcal{H}, \mathcal{R}) = \frac{1}{n-k} \times \sum_{i=1}^{n-k} (|fr(\mathcal{R}, i, i+k) - fr(\mathcal{H}, i, i+k)|) \tag{5.3}$$

où n correspond au nombre de phrases du texte concerné et $fr(\mathcal{S}, i, j)$ représente une fonction retournant le nombre de frontières de $\mathcal{B}$ existant entre les phrases u_i et u_j dans la segmentation $\mathcal{S} = (\mathcal{U}, \mathcal{B})$:

$$fr(\mathcal{S}, i, j) = |\{b \in \mathcal{B} | i \leq b < j\}| \tag{5.4}$$

En réalisant une rapide analyse de la mesure, on s'aperçoit que la taille k de la fenêtre utilisée joue un rôle très important dans la mesure. En effet, si d correspond à la distance qui sépare une frontière de sa position dans la référence (en nombre de phrases), on a alors :

- Si $d \in [1, k/2[$, alors la pénalité résultant du décalage est moins importante que si la frontière concernée n'avait pas été détectée du tout ;
- Si $d = k/2$, alors la pénalité résultant du décalage est égale à la pénalité qui aurait été donnée si la frontière concernée n'avait pas été détectée du tout ;

- Si $d \in]k/2, k]$, alors la pénalité est plus importante que si la frontière n'avait pas été détectée mais plus faible que si la mesure avait considéré deux erreurs (une frontière manquante et une frontière abusive) ;
- Si $d > k$, alors la mesure considère deux erreurs (une frontière manquante et une frontière abusive).

La taille de fenêtre utilisée détermine alors la tolérance de la mesure par rapport aux erreurs. Pevzner et Hearst conseillent de fixer cette valeur k de manière à ce qu'elle corresponde à la moitié de la taille moyenne des segments de la référence [Pevzner and Hearst, 2002] :

$$k = round(\frac{1}{2} \times \frac{n}{|\mathcal{B}_{\mathcal{R}}| + 1}) \tag{5.5}$$

avec $round(x)$ une fonction retournant l'arrondi de x à l'entier le plus proche et n correspondant au nombre de phrases du texte concerné. De cette manière, en supposant que tous les segments de la référence font à peu près la même taille, les frontières mal placées sont en fait toutes considérées comme des décalages dont on peut évaluer l'importance selon leur distance à la frontière la plus proche.

Pevzner et Hearst ont montré que cette mesure permettait de pallier aux biais relatifs à la *Pk-mesure*. Plus précisément, elle permet de gagner en stabilité face aux variations de taille des segments et se révèle aussi sévère avec les frontières manquantes qu'avec les frontières abusives.

5.4 Vers une segmentation globale et cohérente des textes

Tel que l'on a pu le voir en section 5.2, les méthodes statistiques de segmentation fondent la détection des ruptures thématiques sur des critères locaux. La plupart de ces méthodes déterminent des zones au sein desquelles des mesures de cohésion peuvent être calculées. Seules les relations de proximité entre unités (phrases ou paragraphes) d'une même zone de texte sont alors considérées.

L'efficacité de ces méthodes semble dépendre fortement de la taille des zones considérées (longueur maximale d'une chaîne lexicale, taille de la *fenêtre de calcul*, nombre de paires voisines considérées, etc ...). Dans le cas de zones trop petites, l'algorithme risque d'ignorer un certain nombre d'indices de cohésion. Dans le cas de zones trop grandes, il risque de considérer des répétitions de termes qui ne de-

vraient pas être associées puisqu'utilisées dans des contextes différents [9]. Cependant, il est difficile de déterminer la taille de ces zones puisqu'aucune information sur la structure du texte n'est disponible *a priori*.

Par ailleurs, de telles approches rendent impossible la prise en compte du texte dans sa globalité, particulièrement pour ce qui concerne le degré de granularité utilisé pour l'expression des différents thèmes abordés. Nous pensons que le fait de restreindre les calculs à des zones de texte peut avoir des effets négatifs sur la qualité et la cohérence de la segmentation obtenue : de la même manière que le partitionnement d'un ensemble de données en classes ne peut être réalisé efficacement à partir d'un échantillon restreint des données, la totalité des relations inter-phrases (ou inter-paragraphes) peut être nécessaire pour distinguer les différents thèmes du texte. Afin d'obtenir une segmentation cohérente, la détermination d'une frontière doit avoir des effets sur le processus de segmentation dans sa globalité.

Afin de pallier à ces limites observées, nous avons cherché à nous tourner vers une méthodologie de segmentation qui puisse considérer les textes de manière plus globale que les approches existantes. Cela nous a conduit à considérer les propositions faites dans [Bellot, 2000], où il est proposé de réaliser un repérage préliminaire de l'ensemble des thèmes du document avant d'y appliquer un processus de segmentation. Ayant souligné la dualité existant entre clustering et segmentation thématique, Patrice Bellot propose d'utiliser les groupes d'unités formés par un processus de clustering pour en déduire les segments thématiques du texte : si deux unités adjacentes ne sont pas contenues par le même cluster, c'est alors qu'elles correspondent à un thème différent et donc qu'elles doivent être séparées dans deux segments différents. Considérant que les documents peuvent opérer des changements thématiques qui ne soient que temporaires (tels que des digressions), la méthode proposée présente l'avantage de produire une segmentation dans laquelle il est possible de relier les segments de même thème. Néanmoins, pour l'utilisation que l'on souhaite faire des segments thématiques obtenus (c'est à dire, leur considération dans des processus de recherche d'information), le découpage thématique proposé nous semble quelque peu inadapté : le fait qu'une phrase donnée soit contenue par un cluster différent de celui auquel appartiennent les phrases qui l'entourent conduit à la production de trois segments distincts. Les structures discursives des documents pouvant être très diverses et une partie d'un texte liée à une thématique particulière étant susceptible de contenir des phrases, telles que des incises, qui en dévient quelque peu, ce cas risque d'être

9. Ce biais est amplifié lors de l'utilisation de méthodes de segmentation par chaînes lexicales, dans lesquelles le fait que les termes soient considérés individuellement ne permet pas de prise en compte de leur contexte d'occurrence. Prenons l'exemple de la configuration décrite par la figure 5.1, où les lettres de A à H correspondent aux unités du texte et les lignes horizontales à des chaînes lexicales créées. Dans cet exemple, la méthode ne détermine pas de frontière thématique entre D et E puisque le nombre de chaînes n'y est pas minimal. Néanmoins, on est en droit de s'interroger sur la signification des chaînes lexicales présentes entre ces deux unités : les couples d'unités (C,D) et (E,F) ne semblent en effet n'avoir aucun terme en commun (aucune chaîne ne relie C ou D à E ou F). Si l'on ne les sépare pas, c'est uniquement grâce aux unités qui les entourent. Si les termes qui permettent de les relier sont employés dans des contextes différents, la détermination d'une frontière entre les deux couples aurait pu être appropriée.

fréquemment observé, conduisant alors à une sur-segmentation du texte [10]. Certes, les liaisons existant entre les segments permettent de faire apparaître l'organisation générale du discours, mais ce type de segmentation risque d'être peu utilisable pour des applications ultérieures, telles que la recherche de passages correspondant à une requête par exemple (approches dites de *Passage Retrieval*), la structure résultant de ce processus de segmentation pouvant alors s'avérer trop complexe. Par ailleurs, la segmentation produite dépend directement du mécanisme de regroupement utilisé, alors qu'aucune méthode de clustering existante ne peut être considérée comme absolument fiable.

Néanmoins, étant convaincus qu'un clustering préliminaire peut permettre d'adopter une vision plus globale du texte et des relations qui régissent l'organisation du discours, nous avons proposé une première méthode de segmentation, appelée *ClassStruggle*, qui, tel que préconisé dans [Bellot, 2000], s'appuie sur un clustering préliminaire des phrases du texte pour en découvrir les thèmes principaux. Afin d'améliorer la robustesse du système, nous avons cherché à donner une certaine flexibilité au processus de déduction des segments à partir des clusters formés. Considérant que le processus de segmentation doit, outre les proximités thématiques des unités, prendre en compte une notion de distance dans le texte, nous proposons de faire évoluer les clusters thématiques en fonction de l'agencement des phrases dans le texte. Nous présentons cette méthode plus en détail dans la section suivante.

Tel que mentionné précédemment, Salton définit un segment thématique comme une partie de texte qui présente à la fois une forte cohésion interne et une grande dissimilarité avec les parties de texte adjacentes [Salton *et al.*, 1996]. Ces deux facteurs, cohésion interne et dissimilarité entre segments adjacents (qui peuvent être rapprochés des critères de cohésion et de séparation des clusters, voir section 2.1), peuvent constituer deux critères importants à optimiser. Or, on peut observer que dans la plupart des approches existantes, la détermination des frontières thématiques se fait de manière séquentielle, ce qui réduit considérablement les capacités d'optimisation de ces critères. En effet, la détermination d'une frontière étant réalisée avant que le segment qui la suit ne soit encore entièrement délimité, il est alors impossible de calculer la proximité entre segments adjacents. La cohésion interne des segments, quant à elle, n'est optimisée que localement et risque alors de ne pas correspondre à un optimum global. Dans des travaux plus récents [Kehagias *et al.*, 2003; Ji and Zha, 2003], les approches proposées cherchent à optimiser la cohésion interne des segments de manière globale en utilisant des algorithmes de programmation dynamique. Néanmoins, dans ces approches, la prise en compte de la dissimilarité entre segments adjacents s'avère problématique puisque, tandis que le calcul de la cohésion interne de tous les segments possibles d'un texte peut paraître raisonnable en terme de temps de calcul, il ne semble pas envisageable de considérer l'ensemble

10. Si le document auquel on applique le processus de segmentation est relativement homogène, il est même possible d'obtenir autant de segments que ce document ne contienne de phrases.

des paires de segments possibles pour calculer leur proximité [11].

Ces observations nous ont conduit à concevoir une approche permettant une meilleure prise en compte de la cohésion interne des segments et de la dissimilarité entre segments adjacents. Basé sur l'évaluation de propositions de segmentation, *SegGen* considère le texte dans son ensemble pour en déduire une segmentation "optimale". Les frontières ne sont pas déterminées les unes après les autres, ce qui permet la mesure des critères de segmentation proposés par Salton. Ne disposant d'aucune information sur la structure du texte ni sur le nombre de frontières à déterminer *a priori*, la taille de l'espace de recherche induit nous a conduit à employer des techniques issues du domaine de l'optimisation combinatoire. L'utilisation d'un algorithme génétique dédié à l'optimisation multi-objectifs, le *Strength Pareto Evolutionary Algorithm* présenté en section 4.3.2, nous a semblé tout à fait adaptée à notre problème de segmentation qui peut alors s'apparenter à un problème d'optimisation à deux objectifs (cohésion interne et dissimilarité entre segments adjacents). Cette approche de segmentation est présentée en section 5.4.2.

5.4.1 ClassStruggle : mise en concurrence de groupes thématiques

La méthode *ClassStruggle* que nous proposons commence par appliquer un processus de clustering aux phrases du texte pour en découvrir les thèmes principaux, ou du moins, faire émerger des tendances de rapprochement thématique entre phrases selon une mesure de similarité donnée (la mesure *Cosine* du modèle vectoriel, voir section 1.3.2) . La méthode de clustering utilisée dans un premier temps est la méthode *Single Pass* décrite par l'algorithme 2.1 (section 2.3), avec une valeur du seuil minimal φ, paramètre que requiert cet algorithme, fixée de telle sorte qu'elle corresponde à la moyenne des distances entre chaque couple de phrases du texte. La distance $\delta(u_i, C_j)$, entre une phrase u_i et un cluster C_j, correspond à la moyenne des distances entre la phrase u_i et les phrases contenues par le cluster C_j. En fin de processus, les clusters ne contenant qu'une seule phrase sont supprimés et les phrases qu'ils contiennent ré-affectées au cluster qui leur est le plus proche. Cette méthode est loin d'être optimale puisque les clusters résultants dépendent fortement de l'ordre de traitement des phrases [Zamir and Etzioni, 1998]. Néanmoins, elle nous permet d'obtenir des groupes de phrases sans avoir à spécifier un nombre de groupes à produire (qui peut donc varier pour chaque texte), ce nombre de clusters dépendant des relations existant entre les phrases du texte. Par ailleurs, l'utilisation d'un tel processus, qui produit des groupes thématiques très approximatifs, nous permettra de rendre compte de la flexibilité du processus de segmentation par rapport aux

11. Dans [Malioutov and Barzilay, 2006], les auteurs tentent de contourner ce problème en calculant la similarité de chaque segment possible avec le reste du texte plutôt que de considérer chaque paire de segments possibles, ce qui réduit considérablement la complexité. Cependant, l'approche proposée ne semble pas obtenir des résultats significativement meilleurs que ceux des méthodes existantes. Cela peut s'expliquer par le fait que, malgré une prise en compte des distances entre phrases dans le texte pour calculer leur similarité, une telle approche ne permet pas une considération réelle de la linéarité du discours.

clusters utilisés. Dans nos expériences, une moyenne de 12 clusters par document de 100 phrases et de 7 clusters par document de 50 phrases ont été produits.

Alors que le principe de la plupart des méthodes statistiques est de s'appuyer sur la distribution des termes dans le texte, *ClassStruggle* se fonde sur la distribution des occurrences des membres de chacun des clusters qu'il manipule. L'objectif est de définir un processus permettant d'introduire un critère de proximité spatiale (positions dans le texte) dans des clusters rendant uniquement compte de proximités thématiques. Dans le but d'obtenir des groupes qui contiennent uniquement des phrases appartenant à un même développement thématique, chaque phrase est amenée à transiter d'un cluster à l'autre, en fonction des clusters de son voisinage (phrases adjacentes). Les phrases contenues par un même cluster sont alors considérées comme membres d'un même segment thématique, et des frontières sont finalement déterminées entre les phrases adjacentes appartenant à des clusters différents.

L'algorithme 5.1 décrit le fonctionnement général de *ClassStruggle* : à chaque itération du processus, l'objectif est de déterminer quel est le cluster dominant sur les différentes zones du texte (ce qui correspond à déterminer la thématique dominante sur les différentes parties de texte), afin d'y faire transiter les phrases correspondantes (phrases de la zone concernée) lors de l'itération suivante. Le processus se base alors sur un calcul du potentiel d'appartenance $A(C_i, j)$ de chaque phrase u_j à chaque cluster C_i, chaque phrase étant finalement affectée au cluster pour lequel elle obtient le meilleur potentiel d'appartenance :

$$A(C_i, j) = Prox(C_i, C(u_j)) + \beta \times Ag(C_i, j-1) + \beta \times Ad(C_i, j+1) \quad (5.6)$$

$$Ag(C_i, j) = Prox(C_i, C(u_j)) + \beta \times Ag(C_i, j-1) \quad (5.7)$$

$$Ad(C_i, j) = Prox(C_i, C(u_j)) + \beta \times Ad(C_i, j+1) \quad (5.8)$$

où $C(u_j)$ correspond à une fonction retournant le cluster auquel la phrase u_j appartient et $Prox(C_i, C_j)$ correspond à la proximité thématique des clusters C_i et C_j (formule 5.10). Le potentiel d'appartenance d'une phrase u_j à un cluster C_i dépend alors de la proximité du cluster contenant la phrase u_j avec le cluster concerné C_i, ainsi que d'un degré de popularité / représentativité thématique du cluster C_i dans la zone de texte concernée (traduit par la proximité de ce cluster C_i avec les clusters contenant les phrases adjacentes à u_j, $Ag(C_i, j-1)$ pour les phrases précédant u_j et $Ad(C_i, u_{j+1})$ pour celles suivant u_j dans le texte $\mathcal{U}$) [12]. Ces deux scores $Ag(C_i, j-1)$ et $Ad(C_i, u_{j+1})$ sont pondérés par un coefficient de propagation β qui détermine l'influence du voisinage sur l'appartenance d'une phrase à un cluster.

La formule 5.6 peut se réécrire, peut être plus simplement, de la manière suivante :

$$A(C_i, j) = \sum_{l=1}^{k} \beta^{|j-l|} \times Prox(C_i, C(u_l)) \quad (5.9)$$

12. Notons que $\forall i \in \{1, \ldots, k\}, Ag(C_i, 0) = 0$ et $\forall i \in \{1, \ldots, k\}, Ad(C_i, n+1) = 0$.

Algorithme 5.1 : Algorithme de la méthode de segmentation ClassStruggle

Données :
Une séquence $\mathcal{U}$ de n unités (ou phrases) $u_1, u_2, \ldots, u_n$,
Un ensemble $\mathcal{C}$ de k clusters $C_1, C_2, \ldots, C_k$ issus du clustering initial des phrases,
Deux paramètres α et β permettant de régler l'influence de la proximité spatiale.
Résultat :
Une liste de frontières $\mathcal{B}$ de m frontières $b_1, b_2, \ldots, b_m$.

début
  $\mathcal{B} = \{\}$; $m = 0$; $best = 0$; $stop = 0$;
  $\mathcal{C}' =$ ensemble de k clusters $C'_1, C'_2, \ldots, C'_k$ vides;
  /* Évolution des clusters */
  tant que $stop = 0$ **faire**
    $stop = 1$;
    pour chaque $C_i \in \mathcal{C}$ **faire**
      pour chaque $u_j \in \mathcal{U}$ **faire**
        Calcul du potentiel d'appartenance $A(C_i, j)$ de la phrase u_j au cluster C_i (formule 5.6 avec les paramètres α et β);
      fin
    fin
    /* Affectation des phrases aux clusters dominants */
    pour chaque $u_j \in \mathcal{U}$ **faire**
      $best = \text{argmax}_{i \in \{1,\ldots,k\}} A(C_i, j)$;
      $C'_{best} = C'_{best} \cup \{u_j\}$;
    fin
    si $\mathcal{C} \neq \mathcal{C}'$ **alors**
      $\mathcal{C} = \mathcal{C}'$;
      $\mathcal{C}' =$ ensemble de k clusters $C'_1, C'_2, \ldots, C'_k$ vides;
      $stop = 0$;
    fin
  fin
  /* Segmentation de $\mathcal{U}$ entre phrases appartenant à des clusters différents */
  Soit $C(u_j)$ une fonction qui retourne le cluster contenant la phrase u_j;
  pour $j = 2$ *à* n **faire**
    si $C(u_j) \neq C(u_{j-1})$ **alors**
      $m = m + 1$;
      $b_m = j - 1$;
      $\mathcal{B} = \mathcal{B} \cup \{b_m\}$;
    fin
  fin
fin

Réécrite de cette manière, la formule fait apparaître une notion de distance au signal [13] : le bénéfice qu'un cluster retire de la présence de l'un de ses membres (ou d'un membre qui lui est thématiquement proche) dans une zone de texte donnée est atténué au fur et à mesure que l'on s'éloigne de ce signal (puisque $\beta \in [0, 1]$).

La proximité thématique $Prox(C_i, C_j)$ entre deux clusters C_i et C_j pourrait être réduite à $Sim(C_i, C_j)$, la similarité moyenne entre phrases des deux clusters selon la mesure *Cosine* [14], mais cela risque de poser problème lorsque l'on considère le potentiel d'appartenance d'une phrase au cluster qui la contient déjà puisque dans ce cas $Prox(C_i, C(u_j)) = Prox(C_i, C_i)$, ce qui est égal à 1 si l'on fixe $Prox(C_i, C_j) = Sim(C_i, C_j)$ (puisque les phrases sont les mêmes dans les deux clusters comparés). Cela donnerait trop d'importance au cluster qui contient la phrase considérée, la valeur 1 étant généralement bien supérieure aux scores de similarité entre clusters distincts. Dans ce cas, le contexte n'influerait pas suffisamment les calculs et les clusters ne changeraient jamais. Nous avons donc choisi de remplacer cette valeur par la similarité maximale existant entre deux clusters distincts pondérée par un coefficient α à déterminer empiriquement. La proximité $Prox(C_i, C_j)$ entre deux clusters C_i et C_j est alors donnée par :

$$Prox(C_i, C_j) = \begin{cases} Sim(C_i, C_j) \quad \text{Si } i \neq j, \\ \\ \alpha \times max_{\{(x,y)\in\{1,\ldots,k\}^2, x\neq y\}}(Sim(C_x, C_y)) \quad \text{Sinon.} \end{cases} \tag{5.10}$$

Le bénéfice tiré par un cluster lorsqu'il rencontre un de ses membres dépend alors du paramètre α qui peut être considéré comme un facteur d'intensité du signal, en opposition avec β qui correspond à un degré de propagation de ce signal.

Une fois les différents potentiels d'appartenance des phrases aux clusters établis, l'algorithme les utilise pour réaffecter les phrases dans les clusters (chaque phrase u_j est affectée au cluster C_i avec lequel elle possède le plus fort score $A(C_i, j)$). Le processus se poursuit (en recalculant les potentiels et réaffectant les phrases aux clusters) tant que l'on n'a pas atteint une configuration stable. Cette réitération du processus permet de répercuter les transformations de clusters sur les calculs et ainsi, pour certains groupes, d'établir clairement leur dominance sur des zones de texte données. Les segments obtenus peuvent alors être suffisamment étendus pour définir de véritables passages thématiquement homogènes malgré les nombreuses irrégularités du texte.

Lors des expérimentations décrites dans la suite de cette section, nous avons observé une convergence rapide de l'algorithme dans la plupart des cas, le nombre maximal d'itérations enregistré étant de 15 itérations. La qualité du clustering initial, le degré de prise en compte du contexte (paramètre β) et la taille des textes à

13. Notion qui peut se rapprocher des chaînes pondérées de [Sitbon and Bellot, 2005], à ceci près que les scores des chaînes dépendent des membres des clusters et pas uniquement à des occurrences de termes.

14. Nous n'utilisons néanmoins pas le facteur IDF employé dans la formulation réalisée en section 1.3.2, Hearst ayant constaté que l'utilisation de ce facteur tendait à réduire quelque peu les performance des algorithmes de segmentation thématique [Hearst, 1994].

segmenter paraissent légèrement influencer le nombre d'itérations nécessaires pour atteindre un ensemble de clusters stable, mais les différences ne sont pas réellement significatives. Néanmoins, il peut probablement survenir que, dans de très rares cas, l'algorithme entre dans une boucle infinie, deux clusters se succédant sur un territoire donné, prenant le pas l'un sur l'autre à tour de rôle. Il est alors possible de définir un nombre d'itérations maximal pour éviter ce genre de cycle, qui reste, selon nos observations, très peu fréquent.

La figure 5.3 illustre un exemple de l'application de la méthode *ClassStruggle* sur un texte composé de 10 phrases (de A à J) extraites de deux articles de journaux différents, les phrases de A à G appartenant à un premier article, les phrases de H à J à un autre. L'application de la méthode de clustering initiale sur ce texte a permis d'identifier trois groupes de phrases C_1, C_2 et C_3. Étant donné que la phrase C possède un certain nombre de termes en commun avec les phrases H, I et J, le processus de clustering les a regroupés dans un même cluster. Les phrases D et E, quant à elles, sont très fortement similaires. Elles ont donc été affectées à un cluster différent des autres phrases. Les potentiels d'appartenance des différentes phrases

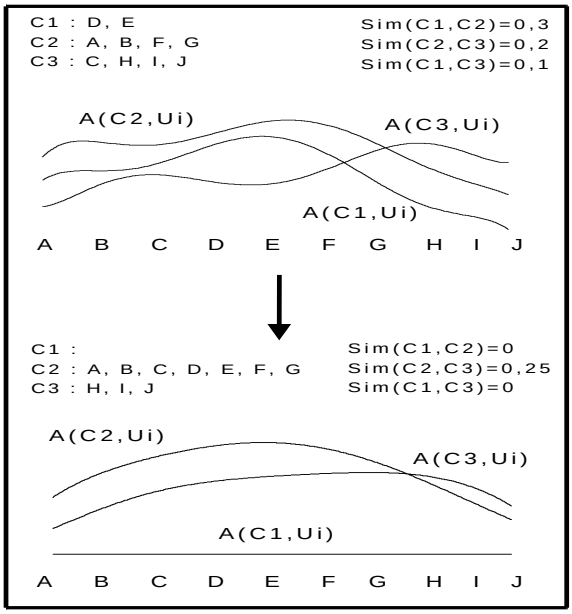

FIGURE 5.3 – Fonctionnement de ClassStruggle

aux trois clusters sont représentés par des courbes dont la hauteur correspond au score obtenu par le couple $cluster, phrase$ correspondant (selon l'équation 5.9). L'algorithme *ClassStruggle* atteint un état stable après la première itération, le cluster C_2 ayant pris le pas sur les autres sur le début du document (de A à G, ce qui correspond au premier article) et le cluster C_3 se trouvant dominant sur la fin du texte (c'est à dire de H à J, le deuxième article). Le cluster C_1 est quant lui complètement

dominé et même vidé de tous ses éléments. La méthode détermine donc finalement une frontière entre les phrases G et H, ce qui correspond bien à la séparation entre les deux articles de journaux. Notons qu'une segmentation directement déduite du clustering aurait conduit à la production de cinq segments : AB, C, DE, FG et HIJ. L'introduction de notions de proximité spatiale des phrases dans les clusters a permis de passer outre les petites irrégularités du discours pour ne s'intéresser qu'aux tendances majeures du texte et déterminer les ruptures thématiques les plus importantes.

Les deux paramètres α et β ont une grande importance puisque ce sont eux qui déterminent le degré de flexibilité du processus de segmentation par rapport au clustering initial. Nous présentons donc maintenant quelques expérimentations réalisées dans le but de trouver les réglages optimaux pour ces deux paramètres. Essayons dans un premier temps d'encadrer ces valeurs :

- Pour pouvoir converger vers une segmentation stable, il faut que le bénéfice tiré par un cluster lorsqu'il rencontre l'un de ses membres (une phrase qu'il contient) soit supérieur à ce que les autres clusters gagnent sur cette même phrase. La valeur de α doit donc être supérieure à 1 pour permettre aux clusters d'obtenir le meilleur score $Prox(C_i, C(u_j))$ si $u_j \in C_i$;
- Pour donner au contexte la possibilité d'influer sur l'appartenance des phrases aux clusters, il ne faut pas que α soit trop élevé. Nous nous limiterons donc à $\alpha = 2$ comme valeur maximale ;
- Pour que le contexte ait une quelconque influence, il faut que la valeur du coefficient de propagation β soit supérieure à 0 (si $\beta = 0$, la segmentation est directement déduite des clusters initiaux) ;
- Le coefficient β doit absolument être inférieur à 1 puisque, avec une telle valeur, un cluster risque de dominer tous les autres de manière permanente, prévenant la détermination de quelque frontière que ce soit (le contexte ayant trop d'importance, le thème possédant le plus de représentants dans le texte risque d'absorber les autres clusters).

Afin de déterminer les valeurs "optimales" de ces deux paramètres, nous avons alors testé toutes les combinaisons (α, β) entre $(1, 0)$ et $(2, 1)$ en faisant varier les valeurs d'un pas de 0.05. La figure 5.4 décrit les résultats de ces expérimentations sur le corpus $AP(100, 4)$ en terme de WindowDiff, de rappel, de precision et de nombre de segments obtenus, chaque courbe représentant les scores obtenus pour une valeur donnée de α, avec β variant entre 0 et 1 (chaque point correspond à la moyenne des scores obtenus sur 350 documents). La courbe de chaque graphe dont les points sont marqués correspond aux résultats obtenus avec la valeur $\alpha = 1.25$.

Rechercher une valeur optimale pour α selon la valeur de β revient à trouver un équilibre entre respect des clusters initiaux et prise en compte du contexte des phrases. Avec de faibles valeurs de β (c'est à dire lorsque le contexte n'a que peu d'influence), les valeurs de α les plus faibles paraissent être les meilleures et lorsque β augmente, la méthode fonctionne mieux avec des valeurs de α légèrement plus

élevées. Les valeurs de α supérieures à 1.25 donnent trop d'importance au cluster contenant la phrase courante et entrainent alors une nette détérioration des résultats. Néanmoins, α semble n'avoir qu'une faible influence sur les résultats.

Les différentes courbes restent stationnaires jusqu'à la valeur $\beta = 0.5$. À partir de cette valeur, le contexte a suffisamment de poids pour influencer la détermination des clusters dominants. Selon les scores de WindowDiff et de précision, $\beta = 0.8$ per-

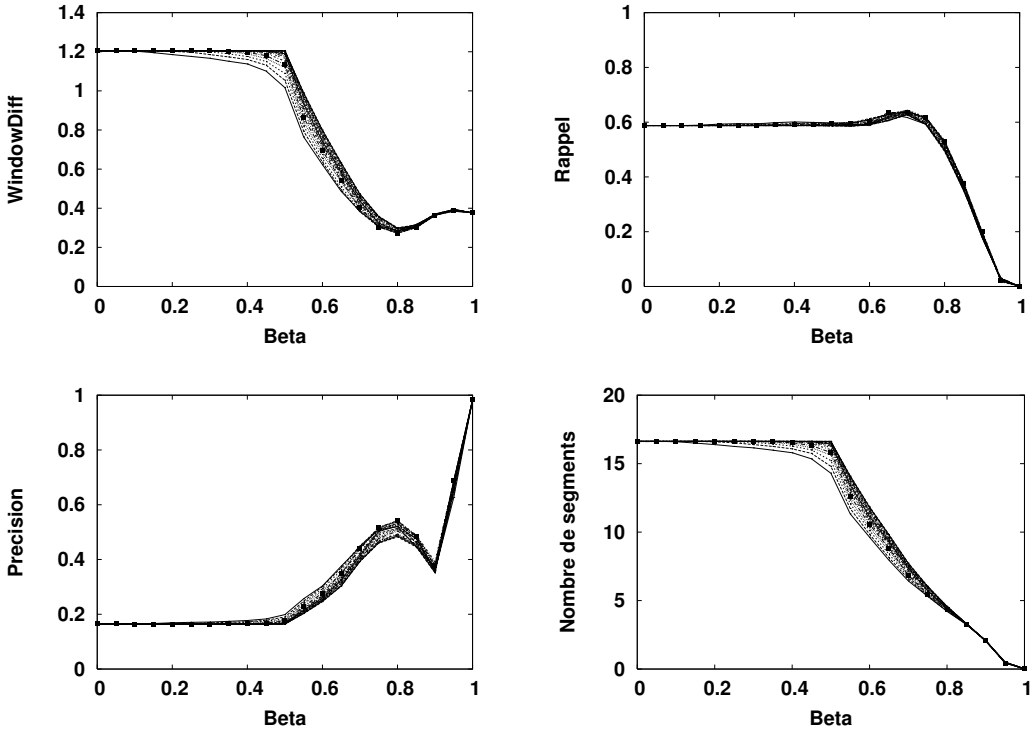

FIGURE 5.4 – Réglages des paramètres α et β de ClassStruggle

met l'obtention des meilleurs résultats. Cependant, une valeur plus faible entraîne un meilleur rappel. Cela est dû au fait qu'une plus grande importance donnée au contexte tend à réduire le nombre de segments. Malgré une diminution du nombre de segments, la précision chute pour $0.8 \leq \beta \leq 0.9$, ce qui dénote une détérioration évidente de la qualité de la segmentation (la précision devrait augmenter avec la diminution du nombre de segments). Les clusters dominants ont, avec de telles valeurs, trop d'impact sur leur entourage et empiètent sur le "territoire" des autres clusters. Ainsi, les valeurs optimales pour β sont situées entre 0.7 et 0.8, selon la fréquence de segmentation désirée. Sur cet intervalle, $\alpha = 1.25$ permet l'obtention des meilleurs résultats. Par ailleurs, le fait que la valeur optimale de β soit nettement supérieure à 0 souligne l'intérêt significatif de notre approche par rapport à un processus déduisant directement ses frontières des clusters de phrases initialement

formés.

5.4.2 SegGen : optimisation multi-objectifs des segments

La segmentation peut être considérée comme un problème d'optimisation multi-objectifs visant à maximiser deux critères : la cohésion interne des segments et la dissimilarité entre segments adjacents. Ces deux critères s'opposent puisque, lorsque le nombre de frontières augmente, la cohésion interne tend à croître alors que la dissimilarité entre segments adjacents tend à diminuer. Afin d'optimiser ces deux critères simultanément, sans favoriser l'un par rapport à l'autre, et en gardant un niveau de diversité des solutions permettant d'atteindre des compromis efficaces, nous utilisons le performant "Strength Pareto Evolutionary Algorithm" [Zitzler, 1999] présenté en section 4.3.2. Réalisant l'application directe de cet algorithme, nous n'en présentons ici que les aspects et détails spécifiques à notre problème.

Représentation du problème

Étant donné un texte composé de n phrases, les individus sont représentés par des vecteurs binaires $\vec{x}$ de $(n-1)$ éléments x_i ; $x_i = 1$ indique qu'une frontière existe entre les phrases i et $i+1$. Ce codage, qui représente exactement la traduction d'un ensemble de frontières $\mathcal{B}$ en vecteur binaire de taille fixe, s'avère très pratique lors de l'application d'opérateurs génétiques. Les vecteurs des individus ont en effet tous la même taille quel que soit le nombre de frontières pour lequel ils codent, ce qui simplifie considérablement les croisements d'individus.

Étant donnés les critères $C(\vec{x}) \in [0,1]$ (cohésion des segments de l'individu) et $D(\vec{x}) \in [0,1]$ (dissimilarité entre segments adjacents), notre problème d'optimisation consiste à approcher l'ensemble des maximiseurs $\mathcal{M}$ des deux objectifs :

$$\begin{aligned} \mathcal{M} = \Big\{ & \vec{x} \in \{0,1\}^{n-1} \mid \forall \vec{x'} \in \{0,1\}^{n-1}, \\ & \vec{x} \neq \vec{x'} \Rightarrow \Big(C(\vec{x}) > C(\vec{x'}) \ \vee \ D(\vec{x}) > D(\vec{x'}) \\ & \qquad \vee \ \big(C(\vec{x}) = C(\vec{x'}) \ \wedge \ D(\vec{x}) = D(\vec{x'}) \big) \Big) \Big\} \end{aligned} \tag{5.11}$$

Le critère de cohésion $C(\overrightarrow{x})$, pour un individu donné $\overrightarrow{x}$, peut être évalué en considérant les moyennes des similarités [15] entre phrases de chacun de ses segments :

$$C(\overrightarrow{x}) = \frac{1}{nbseg} \times \sum_{i=1}^{nbseg} \frac{SumSim(seg_i)}{NbCouples(seg_i)} \tag{5.12}$$

15. Les deux critères à optimiser utilisent des similarités entre phrases calculées au préalable selon la mesure *Cosine*. De la même manière que pour *ClassStruggle*, nous n'utilisons cependant pas le facteur *idf* de la pondération généralement employée (tel que le conseille [Hearst, 1994] lors de calculs de similarités dans un processus de segmentation thématique).

où seg_i correspond au i-ème segment de l'individu, $nbseg$ au nombre de segments qu'il contient, $SumSim(seg_i)$ à la somme des similarités entre toutes les phrases de seg_i et $NbCouples(seg_i)$ au nombre de couples de phrases possibles dans seg_i.

La distribution des similarités dans le texte n'étant pas uniforme, il est possible de trouver de petites zones de texte possédant une cohésion interne très forte. En utilisant la formule (5.12), des individus contenant ces petits blocs de textes ont de grandes chances d'obtenir un score de cohésion élevé. Une normalisation de ce score selon la taille de chaque segment aurait pu être envisagée mais cela aurait probablement induit un certain nombre de biais. La formule (5.13), correspondant à un moyennage global, produit de meilleurs résultats :

$$C(\overrightarrow{x}) = \frac{\sum_{i=1}^{nbseg} SumSim(seg_i)}{\sum_{i=1}^{nbseg} NbCouples(seg_i)} \tag{5.13}$$

En présence de longs segments, le nombre de couples envisagés est plus grand que si tous les segments sont de même taille (par exemple, $C_{n+1}^2 + C_{n-1}^2 > C_n^2 + C_n^2$). Les individus possédant une moindre variation dans la taille de leurs segments sont alors favorisés. Ceci peut être perçu comme un biais mais correspond en fait à favoriser les segmentations réalisant un découpage uniforme.

La similarité d'un segment avec son successeur est calculée grâce à la formule suivante [16] :

$$SimSeg(seg_1, seg_2) = \frac{\sum_{s_j \in \mathcal{P}(seg_1)} \sum_{s_k \in \mathcal{P}(seg_2)} Sim(s_j, s_k)}{|\mathcal{P}(seg_1)| \times |\mathcal{P}(seg_2)|} \tag{5.14}$$

avec seg_i représentant le segment i de l'individu, s_j la phrase j du texte, $\mathcal{P}(seg_i)$ l'ensemble des phrases du segment seg_i et $|\mathcal{P}(seg_i)|$ le cardinal de cet ensemble. Cette formule nous permet d'évaluer la dissimilarité globale entre segments adjacents d'un individu :

$$D(\overrightarrow{x}) = 1 - \left(\frac{\sum_{i=1}^{nbseg-1} SimSeg(seg_i, seg_{i+1})}{nbseg - 1} \right) \tag{5.15}$$

avec $nbseg$ le nombre de segments de $\overrightarrow{x}$.

16. Cette similarité aurait pu correspondre à une similarité calculée directement par un cosinus entre les vecteurs des segments (plutôt qu'en réalisant la moyenne des similarités entre phrases) mais cela aurait impliqué un trop grand nombre de calculs (nouvelles pondérations, nouvelles similarités). Cette solution nous a alors paru plus raisonnable dans un contexte où le nombre d'évaluations risque d'être relativement conséquent.

Génération de la population initiale

Les individus de la population initiale sont générés de manière aléatoire, tout en visant à produire un ensemble d'individus suffisamment hétérogène. Ainsi, afin d'obtenir un degré de diversité suffisant, les frontières de chaque individu de la population initiale sont déterminées en fonction de celles des autres : la probabilité de déterminer une frontière à une position donnée pour un individu est inversement proportionnelle au nombre de fois que cette frontière a déjà été choisie, pour les individus précédemment créés. La taille de la population N est déterminée empiriquement.

Par ailleurs, les algorithmes génétiques ont tendance à converger plus facilement vers des solutions "optimales" lorsque les individus de la population initiale sont déjà des solutions relativement acceptables [Goldberg, 1989]. Nous avons alors expérimenté plusieurs insertions d'individus résultant d'une méthode de segmentation externe. Les résultats concernant ces différentes insertions d'individus dans la population initiale sont présentés et discutés en section 5.4.3.

Opérateurs génétiques

Trois opérateurs sont utilisés dans le processus d'évolution : la sélection, le croisement et la mutation. À chaque génération, l'algorithme sélectionne N individus et en produit N nouveaux afin de maintenir une population de taille fixe. Les individus sont sélectionnés dans $\bar{P} \cup P_t$ par sélection proportionnelle "Roulette Wheel" selon leur valeur d'adaptation [17] (voir section 4.2).

Parmi les individus sélectionnés, tant que le nombre de N nouveaux individus n'est pas atteint, deux individus sont choisis aléatoirement pour en produire deux nouveaux par croisement à un point (voir section 4.2).

Deux opérateurs de mutation sont appliqués : une mutation remplaçant un parent par un individu produit aléatoirement avec une probabilité Pms et une mutation décalant d'une phrase une frontière du nouvel individu avec une probabilité Pmc. Ces deux opérateurs sont complémentaires, puisque le premier permet d'explorer des zones très éloignées des individus envisagés, alors que le second permet plutôt de réaliser un raffinement des solutions déjà rencontrées.

L'algorithme de Zitzler prévoit l'utilisation d'un processus de clustering pour opérer une réduction de l'archive lorsque le nombre d'individus non-dominés devient trop important (voir section 4.3.2). Ainsi, lorsque le nombre de propositions de segmentation appartenant à l'archive dépasse les $3 \times N$ individus, une réduction de l'archive est opérée par utilisation de la méthode de clustering hiérarchique *Group Average* (voir section 2.3.3). Les distances considérées entre individus de l'archive correspondent à une mesure de distance euclidienne (formule 4.3) appliquée aux

17. Les valeurs d'adaptation des individus sont calculées avec les formules de Zitzler, formules 4.4 et 4.5, selon nos deux objectifs de cohésion et de dissimilarité des segments.

scores des deux objectifs de cohésion et de dissimilarité des segments. Lorsque l'algorithme de clustering a atteint un nombre de N groupes, le processus identifie l'élément le plus au centre de chaque groupe. Ces représentants de groupes sont conservés dans l'archive, les autres individus sont tout simplement supprimés.

Extraction d'une solution

Tous les documents ne nécessitant pas le même nombre de générations pour atteindre une segmentation satisfaisante, l'algorithme s'arrête au bout d'un nombre de générations sans évolution significative de la population. $\bar{P}$ constitue, à la fin du processus, l'ensemble des segmentations potentielles d'un texte. Une solution unique doit alors être extraite de cette archive. Le choix de cette solution dépend d'une fonction d'agrégation des deux critères $C(\vec{x})$ et $D(\vec{x})$:

$$Ag(\vec{x}) = C(\vec{x}) + \alpha \times D(\vec{x}) \tag{5.16}$$

La solution extraite est celle obtenant le meilleur score selon cette fonction d'agrégation. Le coefficient α pondère le second objectif par rapport au premier. Il agit sur les caractéristiques de la segmentation finale du texte et permet donc de paramétrer l'algorithme selon le critère que l'on souhaite privilégier et surtout, selon la fréquence de segmentation désirée.

Paramétrage de l'algorithme

La détermination des paramètres d'un algorithme génétique est toujours une tâche complexe en raison des phénomènes stochastiques qu'il met en jeu, ainsi que du très grand nombre de combinaisons de paramètres possibles [Lobo *et al.*, 2007]. Dans le cas de SegGen, cette tâche est d'autant plus complexe que les tests doivent être réalisés sur un large ensemble de textes distincts.

La qualité de la population, notée ci-après $Eval(\bar{P})$, est évaluée selon le score $Ag(\vec{x})$ de l'individu de $\bar{P}$ possédant le meilleur score d'agrégation des critères (formule 5.16) :

$$Eval(\bar{P}) = Max_{\vec{x}\in\bar{P}} Ag(\vec{x}) \tag{5.17}$$

Une étude de différentes archives $\bar{P}$ issues d'exécutions de SegGen sur de multiples textes a montré que le fait de pondérer le critère de dissimilarité par rapport au critère de cohésion d'un coefficient $\alpha = 5$ (formule 5.16) permet la sélection d'un bon individu dans la majorité des cas. Le nombre d'individus $N = 10$, utilisé pour l'initialisation, la sélection et la production de nouveaux individus, apparait de prime abord fournir les meilleurs résultats. Les tests qui suivent utilisent donc ces valeurs.

Dans le but d'ajuster les probabilités de mutation Pms et Pmc, SegGen a été exécuté sur de nombreux textes du corpus $AP(50, 2)$, pour chaque couple (Pms, Pmc) entre $(0, 0)$ et $(1, 1)$, en faisant varier les valeurs d'un pas de 0.2. La figure 5.5 représente l'évolution, au cours des générations, de la moyenne des scores

d'évaluation de la population $Eval(\bar{P})$, enregistrés avec différentes combinaisons de probabilités de mutations [18].

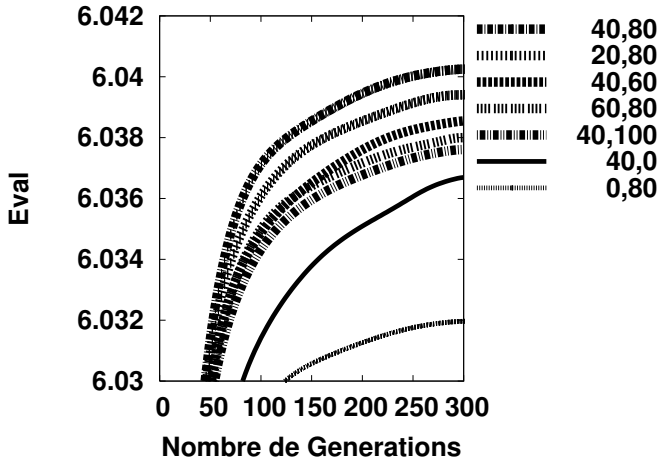

FIGURE 5.5 – Évolution de la population au cours des générations

Cette étude a tout d'abord montré la complémentarité des deux opérateurs de mutation puisque les résultats obtenus avec un seul opérateur sont les plus faibles. Le couple ($Pms = 0.4, Pmc = 0.8$) apparait être le meilleur compromis puisqu'il permet la meilleure convergence de l'algorithme. Par ailleurs, avec ce couple de probabilités, un nombre maximal de 100 générations sans amélioration de la population semble être suffisant pour atteindre de bonnes solutions dans la majorité des cas.

Le paramètre α utilisé dans la fonction d'agrégation des critères semble grandement influencer la segmentation finale. Le critère de cohésion C favorise les individus possédant un grand nombre de frontières, les petits segments ayant plus de chances de posséder uniquement des phrases fortement similaires. Au contraire, le critère de dissimilarité entre segments adjacents D favorise, quant à lui, les individus possédant peu de frontières, puisque les longs segments ont davantage de chances de contenir des phrases éloignées de celles du segment suivant. Le paramètre α a donc une influence sur le nombre de frontières final (voir figure 5.6). Selon les courbes de la figure 5.6, les meilleurs résultats selon le critère WindowDiff sont obtenus avec les valeurs entourant $\alpha = 5$. Néanmoins, une valeur plus élevée permet d'obtenir une meilleure précision et une valeur plus faible un meilleur rappel. Cette valeur doit donc être fixée entre 4 et 6 selon la fréquence de segmentation souhaitée.

Enfin, une étude de la taille de la population P_t a montré que des nombres d'individus supérieurs à $N = 10$ permettent à l'algorithme de converger plus rapidement vers de bonnes solutions (en nombre de générations), mais induisent un

18. Les légendes correspondent aux valeurs Pms, Pmc (multipliées par 100) et sont données dans le même ordre que les scores obtenus en fin d'exécution (la combinaison placée au dessus des autres est la meilleure).

plus grand nombre d'évaluations d'individus. Par exemple, sur le corpus $AP(50, 2)$, l'algorithme a besoin de 150 générations pour atteindre un score d'évaluation de la population $Eval(\bar{P}) = 6.04$ lorsque la taille de la population courante est de $N = 20$ individus, contre 250 générations pour atteindre le même score avec $N = 10$ individus. Cependant, le nombre d'individus générés est plus important dans le premier cas ($150 \times 20 > 250 \times 10$), ce qui implique un plus grand nombre d'évaluations

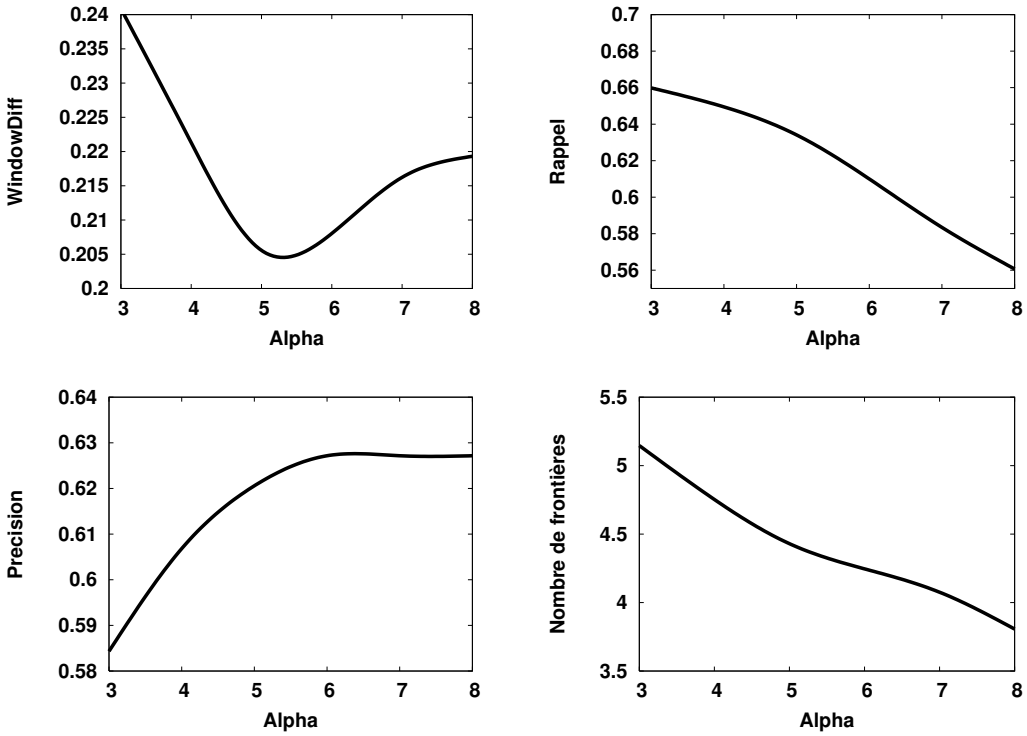

FIGURE 5.6 – Résultats des expérimentations sur le paramètre α de SegGen

d'individus (opération la plus coûteuse du processus). Le problème inverse est observé lorsque $N < 10$, le nombre de générations nécessaires pour atteindre un score d'évaluation donné est bien trop important pour trouver une contre-partie dans le plus faible nombre d'individus évalués à chacune d'entre elles. $N = 10$ semble alors être la taille de population permettant d'observer les meilleures performances.

5.4.3 Évaluation des systèmes

Afin d'évaluer les performances de nos deux algorithmes de segmentation, quatre corpus ont été constitués par concaténation d'articles du corpus AP (voir section

3.2) : $AP(50,2)$, $AP(50,4)$, $AP(100,4)$ et $AP(100,8)$ [19]. À cela viennent s'ajouter deux corpus résultant de la concaténation d'articles du corpus $ZIFF$, notés $ZF(100,4)$ et $ZF(100,8)$, qui permettent d'évaluer la capacité des méthodes à segmenter des textes plus spécifiques [20]. Chacun de ces six différents corpus constitue un ensemble de 350 documents sur lesquels diverses méthodes de segmentation sont appliquées pour déterminer les performances de nos approches.

Les algorithmes *ClassStruggle* et *SegGen* ont été comparés aux méthodes suivantes :

- **Rand** : Étant donné un corpus $T(n,m)$, cette procédure crée aléatoirement m frontières dans chaque texte. Elle représente un point de référence auquel comparer les résultats, présentant le grand avantage de connaître le nombre de frontières à déterminer mais n'utilisant aucun mécanisme pour les positionner au niveau des ruptures thématiques ;
- **TT** : La méthode TextTiling [21] [Hearst, 1994], certainement la plus populaire des méthodes de segmentation, est une extension de la méthode proposée dans [Morris and Hirst, 1991]. Elle est basée sur un concept de fenêtre de calcul qu'elle fait se déplacer sur le texte afin de comparer, en différentes positions, deux blocs de texte adjacents selon différents critères de cohésion lexicale (nombre de mots communs, nombre de mots nouveaux, nombre de chaînes lexicales actives, etc...). Les paramètres principaux de TextTiling, c'est à dire la taille des blocs à comparer (soit la moitié de la taille de la fenêtre) et le pas de déplacement de la fenêtre, ont été étudiés de la même manière que ceux de nos méthodes sur nos corpus d'entraînement. Les meilleurs résultats ont été obtenus avec des blocs de 120 mots (soit une fenêtre de 240 mots) et un pas de déplacement de 20 mots. Les résultats reportés dans cette section ont été obtenus en utilisant ces deux valeurs ;
- **C99** : L'algorithme C99 [22] [Choi, 2000] est, selon de nombreuses études, l'une des méthodes existantes les plus performantes (voir par exemple [Bestgen and Piérard, 2006]). Il détermine, pour chaque couple d'unités textuelles adjacentes, un score correspondant au nombre de couples d'unités voisines (appartenant aux k couples d'unités les plus proches) possédant une similarité inférieure. Il réalise un classement des couples d'unités selon ce score puis dispose les unités sur un plan 2D selon leur position dans le texte (d'une manière similaire à la méthode DotPlotting [Reynar, 2000]). L'algorithme recherche alors les zones du plan les plus denses selon la position des unités dans le classement établi. Les zones qui maximisent cette densité forment les segments du texte.

19. Afin de ne pas biaiser les résultats, ces corpus utilisent des articles différents de ceux utilisés pour le paramétrage des algorithmes.

20. L'application de méthodes de segmentation thématique sur ces deux corpus, dont les textes sont plus homogènes puisque composés d'articles traitant tous d'un même domaine (l'informatique), simule mieux la recherche des ruptures thématiques sur des textes réels.

21. Disponible à l'adresse : www.sims.berkeley.edu/~hearst/

22. Disponible à l'adresse : www.freddychoi.me.uk

Le paramètre principal de C99, la taille du masque k, a été étudié sur les corpus d'entraînement. Les meilleurs résultats ont été obtenus avec une taille de masque $k = 11$, cette valeur est donc celle employée dans les expériences qui suivent ;

– **JI** : Cette méthode est une implémentation de l'algorithme de programmation dynamique présenté dans [Ji and Zha, 2003]. Elle nous paraît constituer l'approche existante la plus performante, surpassant bien souvent la méthode $C99$ en terme de qualité des frontières déterminées. Afin d'augmenter les capacités d'identification des ruptures thématiques les plus importantes du texte, la méthode *JI* commence par supprimer les termes génériques ou uniformément distribués, avant d'appliquer un processus de traitement d'image (*anisotropic diffusion* [Perona and Malik, 1990]) sur la matrice des similarités pour en amplifier les contrastes (et ainsi révéler, de manière plus nette, des zones de texte possédant une forte cohésion lexicale). La méthode cherche ensuite, par un algorithme de programmation dynamique, à minimiser un coût de segmentation global qui dépend de la cohésion interne des segments. Les paramètres principaux de l'algorithme, ceux du processus de traitement d'image, ont été expérimentés sur les corpus d'entrainement. Les meilleurs résultats ont été obtenus au bout de 10 itérations avec un coefficient de conduction de 20 et un coefficient d'évolution λ de 0.2 (valeurs qui ont aussi permis l'obtention des meilleurs résultats dans [Ji and Zha, 2003]). Ces valeurs sont celles utilisées dans les expérimentations qui suivent.

Sept versions de *ClassStruggle* ont été expérimentées :

– $\mathbf{C}_R$: Une version n'utilisant pas de processus de clustering préliminaire mais qui génère les clusters initiaux aléatoirement (12 clusters pour les documents de 100 phrases et 7 pour les documents de 50, c'est à dire autant que les moyennes observées en utilisant la méthode de clustering *Single Pass*) ;

– $\mathbf{C}_{SP(0.7)}$, $\mathbf{C}_{SP(0.75)}$ et $\mathbf{C}_{SP(0.8)}$: Trois versions de ClassStruggle utilisant la méthode de clustering *Single-Pass* (décrite par l'algorithme 2.1) pour générer les clusters initiaux. β est respectivement égal à 0.7, 0.75 et 0.8 ;

– $\mathbf{C}_{ND(0.7)}$, $\mathbf{C}_{ND(0.75)}$ et $\mathbf{C}_{ND(0.8)}$: Trois versions de ClassStruggle qui utilisent un raffinement des classes obtenues par la méthode de clustering *Single-Pass* : la méthode de nuées dynamiques (décrite par l'algorithme 2.2) réaffecte chaque phrase à la classe qui lui est la plus similaire. Ce processus converge vers un état stable au bout de dix itérations en moyenne. Ces trois versions de *ClassStruggle*, utilisant respectivement une valeur de β égale à 0.7, 0.75 et 0.8, permettent d'évaluer l'influence de la qualité des clusters initiaux sur la segmentation finalement obtenue.

Quatre versions de *SegGen*, chacune utilisant un coefficient $\alpha = 5$, ont été expérimentées :

– **S** : L'algorithme *SegGen* n'introduisant pas d'individu résultant d'une segmentation externe dans sa population initiale ;

- $\mathbf{S}_{C99}$: L'algorithme *SegGen* introduisant un individu résultant de $C99$ dans sa populations initiale ;
- $\mathbf{S}_{C_{ND(0.7)}}$: L'algorithme *SegGen* introduisant un individu résultant de $\mathrm{C}_{ND(0.7)}$ dans sa populations initiale ;
- $\mathbf{S}_{Both}$: L'algorithme *SegGen* introduisant deux individus externes dans sa population initiale, l'un provenant de $C99$, l'autre de $\mathrm{C}_{ND(0.7)}$.

La méthode *SegGen* n'étant pas déterministe, nous avons commencé les expérimentations par une évaluation de la marge d'erreur de ses résultats, en calculant l'écart-type des moyennes des scores obtenus sur dix exécutions par corpus. L'écart-type observé pour le critère WindowDiff tourne autour de 0.015, pour le rappel autour de 0.01 et pour la précision autour de 0.012 sur chaque corpus. Ces marges d'erreur paraissent très peu significatives au vu des résultats contenus par les tables 5.1 et 5.2. Il ne semble alors pas utile de relancer *SegGen* de multiples fois sur chaque document pour obtenir des résultats représentatifs des performances réelles de l'algorithme.

Les deux tables 5.1 et 5.2 présentent les résultats obtenus par les différentes méthodes sur les six corpus d'évaluation selon les critères de WindowDiff (W), précision (P), rappel (R) et nombre de frontières détectées (N) (μ représentant la moyenne et σ l'écart-type des résultats sur l'ensemble des textes d'un corpus).

Tel que mentionné en section 5.3.2, le critère *WindowDiff* constitue une meilleure mesure d'évaluation des méthodes de segmentation que les critères classiques de rappel et de precision puisqu'il permet de distinguer les erreurs mineures (décalages de quelques phrases) des erreurs plus importantes (absence ou insertion de frontières) [Pevzner and Hearst, 2002]. Cependant, tel que nous le verrons dans la prochaine section, une évaluation sur ce critère n'a de sens que si les méthodes comparées déterminent un nombre de frontières relativement proche. Une méthode déterminant un plus grand nombre de frontières a en effet une plus grande probabilité d'obtenir un score élevé de WindowDiff (elle s'en trouve alors pénalisée). Ainsi, les scores de *WindowDiff* obtenus par *C99* et $C_{SP(0.8)}$, par exemple, ne peuvent pas être comparés ($C_{SP(0.8)}$ risquerait alors d'être favorisée puisque déterminant généralement moins de frontières que *C99*). Cependant, l'approche $C_{SP(0.7)}$ créant à peu près autant de frontières que *C99*, le fait qu'elle obtienne un meilleur score de *WindowDiff*[23] dénote de la meilleure qualité de ses segmentations. De plus, le rappel et la precision de $C_{SP(0.7)}$ sont clairement supérieurs à ceux de *C99* et *TextTiling* sur tous les corpus. Selon les résultats obtenus (tables 5.1 et 5.2), *ClassStruggle* apparait donc globalement plus performant que les méthodes existantes.

Par ailleurs, les très faibles résultats obtenus par la version de *ClassStruggle* utilisant un clustering aléatoire (C_R) soulignent l'importance du processus de clustering initial. L'algorithme de clustering *Single-Pass* utilisé dans la version originale de *ClassStruggle* est loin d'être optimal. On est alors en droit de supposer que l'uti-

23. Il est à noter que cette mesure correspond à un taux d'erreur. Par conséquent, plus petit est le score obtenu, meilleure est la qualité de la segmentation évaluée.

AP(50,2)	W		P		R		N	
	μ	σ	μ	σ	μ	σ	μ	σ
Rand	0.46	0.21	0.05	0.19	0.05	0.19	2.02	0.98
TT	1.02	0.71	0.26	0.22	0.54	0.38	4.21	1.39
C99	0.89	0.75	0.32	0.22	0.63	0.38	4.22	1.42
JI	0.38	0.38	0.52	0.35	0.62	0.36	2.71	1.48
C_R	0.93	0.90	0.09	0.19	0.16	0.29	3.88	2.76
$C_{SP(0.7)}$	0.43	0.42	0.49	0.30	0.68	0.35	3.39	1.66
$C_{SP(0.75)}$	0.36	0.42	0.54	0.36	0.62	0.38	2.62	1.40
$C_{SP(0.8)}$	0.31	0.35	0.55	0.39	0.52	0.39	2.10	1.20
$C_{ND(0.7)}$	0.40	0.41	0.50	0.30	**0.72**	0.38	3.12	1.43
$C_{ND(0.75)}$	0.33	0.34	0.54	0.34	0.68	0.37	2.79	1.38
$C_{ND(0.8)}$	0.26	0.37	0.58	0.40	0.60	0.36	2.11	1.10
S	**0.22**	0.36	**0.61**	0.39	0.69	0.37	2.44	1.32
S_{C99}	**0.22**	0.37	**0.61**	0.39	0.69	0.37	2.47	1.36
$S_{C_{ND(0.7)}}$	**0.22**	0.36	**0.61**	0.38	0.69	0.37	2.45	1.34
S_{Both}	**0.22**	0.36	**0.61**	0.39	0.69	0.37	2.45	1.34

AP(50,4)	W		P		R		N	
	μ	σ	μ	σ	μ	σ	μ	σ
Rand	0.53	0.14	0.06	0.12	0.06	0.12	4.03	1.45
TT	0.40	0.26	0.49	0.26	0.49	0.26	4.31	1.40
C99	0.38	0.31	0.48	0.23	0.59	0.27	4.94	1.31
JI	0.29	0.20	0.57	0.26	0.64	0.28	4.21	1.55
C_R	0.50	0.39	0.18	0.22	0.18	0.20	4.24	3.16
$C_{SP(0.7)}$	0.29	0.21	0.57	0.27	0.65	0.29	4.67	1.78
$C_{SP(0.75)}$	0.27	0.20	0.60	0.29	0.58	0.30	3.91	1.54
$C_{SP(0.8)}$	0.28	0.17	0.61	0.38	0.49	0.29	3.03	1.37
$C_{ND(0.7)}$	0.27	0.21	0.58	0.28	**0.67**	0.29	4.47	1.62
$C_{ND(0.75)}$	0.23	0.17	0.60	0.27	0.60	0.29	3.96	1.60
$C_{ND(0.8)}$	0.25	0.16	0.63	0.38	0.52	0.31	3.42	1.51
S	0.19	0.17	0.68	0.28	0.60	0.27	3.49	1.40
S_{C99}	0.18	0.18	0.68	0.28	0.64	0.27	3.74	1.51
$S_{C_{ND(0.7)}}$	**0.17**	0.18	0.68	0.28	0.66	0.25	3.71	1.50
S_{Both}	**0.17**	0.18	**0.69**	0.28	**0.67**	0.28	3.75	1.45

AP(100,4)	W		P		R		N	
	μ	σ	μ	σ	μ	σ	μ	σ
Rand	0.55	0.15	0.04	0.09	0.04	0.09	4.20	1.57
TT	0.97	0.66	0.27	0.14	0.59	0.25	9.52	2.33
C99	0.54	0.45	0.40	0.19	0.64	0.26	6.81	2.09
JI	0.35	0.30	0.51	0.28	0.62	0.26	5.62	2.18
C_R	0.79	0.42	0.07	0.19	0.12	0.17	7.07	2.96
$C_{SP(0.7)}$	0.40	0.41	0.48	0.23	0.65	0.27	7.21	2.94
$C_{SP(0.75)}$	0.33	0.29	0.52	0.26	0.60	0.27	5.23	2.29
$C_{SP(0.8)}$	0.29	0.23	0.53	0.28	0.52	0.28	4.37	1.86
$C_{ND(0.7)}$	0.37	0.39	0.50	0.23	**0.68**	0.28	7.12	2.83
$C_{ND(0.75)}$	0.30	0.31	0.53	0.26	0.66	0.27	5.30	2.23
$C_{ND(0.8)}$	0.25	0.25	0.55	0.28	0.53	0.29	4.30	1.82
S	0.20	0.22	0.62	0.29	0.61	0.27	4.19	1.87
S_{C99}	0.20	0.22	0.63	0.29	0.64	0.27	4.48	2.10
$S_{C_{ND(0.7)}}$	**0.19**	0.21	0.65	0.29	0.66	0.28	4.50	2.17
S_{Both}	**0.19**	0.22	**0.66**	0.28	0.66	0.28	4.62	2.01

AP(100,8)	W		P		R		N	
	μ	σ	μ	σ	μ	σ	μ	σ
Rand	0.57	0.10	0.08	0.09	0.08	0.09	8.15	2.30
TT	0.37	0.16	0.47	0.17	0.56	0.18	9.82	2.03
C99	0.33	0.16	0.53	0.17	0.59	0.20	9.15	2.23
JI	0.29	0.17	0.56	0.19	0.61	0.22	8.53	2.52
C_R	0.55	0.14	0.17	0.16	0.14	0.13	7.49	3.03
$C_{SP(0.7)}$	0.30	0.16	0.54	0.18	0.63	0.21	9.61	2.80
$C_{SP(0.75)}$	0.28	0.13	0.57	0.21	0.56	0.22	7.86	2.45
$C_{SP(0.8)}$	0.29	0.12	0.59	0.28	0.48	0.20	6.11	2.26
$C_{ND(0.7)}$	0.29	0.17	0.55	0.19	0.64	0.21	9.44	2.46
$C_{ND(0.75)}$	0.26	0.13	0.60	0.19	0.60	0.20	8.10	2.35
$C_{ND(0.8)}$	0.28	0.14	0.61	0.19	0.51	0.20	6.49	2.33
S	0.22	0.12	0.67	0.21	0.56	0.20	6.50	1.80
S_{C99}	0.18	0.11	0.69	0.19	0.65	0.21	7.75	2.11
$S_{C_{ND(0.7)}}$	0.18	0.12	0.69	0.18	0.66	0.21	7.83	2.21
S_{Both}	**0.17**	0.11	**0.70**	0.16	**0.68**	0.19	7.87	2.02

TABLE 5.1 – Évaluation des méthodes de segmentation sur le corpus AP

lisation de techniques de clustering plus performantes permettrait à *ClassStruggle* de produire des segmentations d'encore bien meilleure qualité. Les résultats obtenus par les versions de *ClassStruggle* appliquant un processus de raffinement des clusters par la méthode de *Nuées Dynamiques* (versions C_{ND}) semblent conforter cette hypothèse puisqu'une amélioration significative de la qualité des segmentations produites est observée.

SegGen, qui considère des propositions de segmentation plutôt que de déterminer des frontières de manière incrémentale, obtient des résultats nettement supérieurs à toutes les autres méthodes sur tous les critères, surtout sur les corpus ne possédant pas un grand nombre de frontières à retrouver (*i.e.*, $AP(50, 2)$, $AP(100, 4)$ et

ZF(100,4)	W		P		R		N	
	μ	σ	μ	σ	μ	σ	μ	σ
Rand	0.55	0.12	0.04	0.07	0.04	0.07	3.82	1.02
TT	0.85	0.62	0.25	0.12	0.58	0.23	8.78	2.08
C99	0.46	0.38	0.39	0.18	**0.63**	0.24	6.26	2.07
JI	0.35	0.39	0.46	0.25	0.61	0.26	5.34	2.20
C_R	0.81	0.39	0.06	0.09	0.10	0.11	6.92	2.83
$C_{SP(0.7)}$	0.43	0.45	0.45	0.25	0.61	0.28	6.31	2.96
$C_{SP(0.75)}$	0.33	0.32	0.47	0.26	0.59	0.28	4.80	1.99
$C_{SP(0.8)}$	0.33	0.28	0.48	0.27	0.51	0.24	4.12	1.81
$C_{ND(0.7)}$	0.42	0.42	0.45	0.24	0.62	0.24	6.35	2.31
$C_{ND(0.75)}$	0.33	0.31	0.47	0.26	0.60	0.27	5.01	2.17
$C_{ND(0.8)}$	0.32	0.28	0.48	0.27	0.53	0.23	4.35	1.75
S	0.20	0.19	0.60	0.28	0.60	0.27	4.12	1.62
S_{C99}	0.20	0.18	0.62	0.30	**0.63**	0.27	4.30	1.57
$S_{C_{ND(0.7)}}$	**0.19**	0.19	0.64	0.30	**0.63**	0.28	4.31	1.63
S_{Both}	**0.19**	0.18	**0.65**	0.29	**0.63**	0.28	4.33	1.60

ZF(100,8)	W		P		R		N	
	μ	σ	μ	σ	μ	σ	μ	σ
Rand	0.57	0.09	0.07	0.07	0.07	0.07	8.04	1.93
TT	0.37	0.16	0.45	0.16	0.53	0.16	9.18	1.95
C99	0.34	0.16	0.50	0.16	0.54	0.17	8.68	1.87
JI	0.30	0.20	0.52	0.21	0.56	0.18	7.91	2.18
C_R	0.56	0.15	0.16	0.16	0.12	0.10	7.33	3.05
$C_{SP(0.7)}$	0.32	0.15	0.50	0.16	0.56	0.20	8.76	2.66
$C_{SP(0.75)}$	0.30	0.18	0.52	0.18	0.54	0.23	6.80	2.30
$C_{SP(0.8)}$	0.29	0.15	0.53	0.19	0.47	0.17	5.93	1.42
$C_{ND(0.7)}$	0.31	0.15	0.50	0.17	0.57	0.15	8.82	2.83
$C_{ND(0.75)}$	0.30	0.17	0.53	0.20	0.55	0.19	7.02	2.46
$C_{ND(0.8)}$	0.29	0.19	0.54	0.20	0.50	0.13	6.13	1.64
S	0.23	0.12	0.66	0.20	0.54	0.17	6.33	1.59
S_{C99}	0.20	0.14	0.66	0.19	0.58	0.18	7.06	1.55
$S_{C_{ND(0.7)}}$	0.20	0.19	**0.67**	0.18	0.60	0.21	7.20	1.52
S_{Both}	**0.19**	0.20	**0.67**	0.20	**0.62**	0.20	7.25	1.54

TABLE 5.2 – Évaluation des méthodes de segmentation sur le corpus ZIFF

$ZF(100, 4)$). *TextTiling* et *C99* semblent rencontrer des difficultés à s'adapter au nombre de frontières à retrouver, la longueur des textes conditionnant largement le nombre de frontières qu'elles détectent. *SegGen* paraît être plus à même de s'adapter à ces variations.

Sur les corpus qui possèdent un grand nombre de frontières à retrouver (*i.e.*, $AP(50, 4)$ et $AP(100, 8)$), *SegGen* semble néanmoins rencontrer quelques difficultés à produire des individus suffisamment segmentés. Notre algorithme converge en effet plus facilement vers des solutions ne possédant que peu de frontières, qui plus est lorsque la taille des textes et donc de l'espace de recherche augmente. Cela est dû au fait que les "bons" individus possédant un grand nombre de frontières sont plus difficiles à atteindre que les autres [24]. L'insertion d'individus provenant d'applications externes dans la population initiale semble corriger ce problème : *C99* et $C_{ND(0.7)}$ tendant toutes deux à déterminer un grand nombre de frontières, l'utilisation de la segmentation qu'ils produisent en tant qu'individu de la population initiale de *SegGen* permet d'aider notre algorithme à orienter sa recherche vers des individus fortement segmentés. Par ailleurs, la qualité de la segmentation sur les textes ne possédant que peu de frontières à retrouver ne semble pas être affectée par cette insertion. L'utilisation conjointe des deux méthodes *C99* et $C_{ND(0.7)}$ pour initialiser la population de *SegGen* améliore encore les résultats, l'algorithme tirant alors partie des "bonnes" caractéristiques de chacune des deux segmentations considérées.

Sur les corpus plus spécifiques, $ZF(100, 4)$ et $ZF(100, 8)$, *ClassStruggle* semble

24. Le nombre de segmentations possibles augmente avec le nombre de frontières déterminées : $m < (n/2) \Rightarrow C_n^m < C_n^{m+1}$. La découverte de bonnes solutions contenant un grand nombre de frontières est alors plus difficile.

perdre un peu de son avantage sur les méthodes existantes, tout particulièrement par rapport à la méthode *JI*. En effet, alors que ses résultats sont légèrement meilleurs que ceux obtenus par *JI* sur les corpus AP (table 5.1), les deux méthodes obtiennent des résultats comparables sur les corpus $ZIFF$ (table 5.2). Sur ces corpus, les phrases des textes étant plus similaires, le processus de clustering rencontre plus de difficultés à distinguer les différentes thématiques abordées. L'utilisation de techniques de clustering plus évoluées, notamment par utilisation de ressources sémantiques spécifiques, devrait permettre d'améliorer les résultats sur ce genre de corpus. *SegGen*, quant à lui, ne semble pas rencontrer réellement plus de difficultés sur ces corpus que sur les autres.

Un test de *Student*[25] a montré que les différences entre les résultats, selon le critère *WindowDiff*, obtenus par notre algorithme de segmentation *SegGen* et ceux obtenus par les autres méthodes de segmentation sont statistiquement significatives avec un degré de certitude de 99%, toutes les valeurs étant supérieures à 2.57. Enfin, les différences en terme de temps de calcul semblent légères : sur le corpus $AP(100, 8)$, corpus impliquant l'espace de recherche le plus étendu, notre algorithme *SegGen* met en moyenne 3 secondes pour produire 100 générations de populations sur un *Pentium 4, 3GHz PC*. La méthode *ClassStruggle*, à l'instar des autres méthodes telles que *C99* ou *TextTiling*, produit une segmentation d'un document en une seconde seulement environ. Selon l'application, la différence de temps de calcul nécessaire entre la méthode *SegGen* et les autres peut sembler significative, mais pour la segmentation d'un corpus de textes avant l'application d'approches telles que celles dites de *Passage Retrieval* par exemple, l'utilisation d'une telle méthode légèrement plus coûteuse paraît tout à fait raisonnable.

5.5 Vers un mode d'évaluation plus équitable

Tel que l'on a pu le voir en section 5.3.2, les méthodes de segmentation thématique sont généralement évaluées par comparaison des frontières qu'elles déterminent avec celles contenues dans des segmentations de référence. Outre les nombreuses difficultés que pose la constitution de telles références (voir section 5.3.1), ce type d'évaluation présente, selon nous, l'inconvénient majeur d'être bien trop rigide face aux segmentations proposées. La segmentation thématique d'un texte est en effet une tâche très subjective et de très nombreuses solutions peuvent se justifier. Les segmentations obtenues par les méthodes dépendent de nombreux facteurs, tels que leur sensibilité face aux changements thématiques ou le point de vue spécifique qu'elles peuvent adopter[26]. Le fait d'évaluer les méthodes sur leurs

25. Le 99% *Student t-test* est basé sur la moyenne, l'ecart-type et le cardinal de deux ensembles de resultats. Il utilise une p-valeur égale à 2.57 pour s'assurer avec un coefficient de certitude de 99% que les différences entre deux collections de resultats sont statistiquement significatives.

26. Par exemple, la méthode proposée dans [Bellot, 2000] adopte un point de vue inhabituel puisqu'elle réalise une segmentation orientée requête.

capacités à retrouver les frontières contenues par une segmentation de référence implique, de la part des méthodes, d'appréhender le texte de la même manière que lors de la détermination des frontières de référence. Dans le cas contraire, les résultats obtenus risquent d'être bien médiocres... Les méthodes ne possédant pas les caractéristiques requises pour se conformer à la segmentation de référence sont largement défavorisées par rapport aux méthodes correctement formatées.

Nous commençons cette section par une analyse de la mesure *WindowDiff* proposée dans [Pevzner and Hearst, 2002]. Cette étude mettant en évidence un certain nombre de biais, notamment concernant l'absence de prise en compte des risques encourus par les méthodes, nous proposons d'adapter la mesure pour corriger ces biais identifiés (qui dépassent la seule rigidité inhérente à l'utilisation d'une référence) et lui donner une certaine flexibilité face aux propositions faites par les méthodes de segmentation à évaluer. Enfin, ces adaptations ne pouvant pas permettre une réelle prise en compte des caractéristiques propres à chaque méthode (puisque la mesure résultante reste tout de même basée sur des segmentations de référence), nous proposons une mesure d'évaluation alternative, basée sur la stabilité des méthodes face à des changements textuels, permettant ainsi une évaluation centrée sur les méthodes elles-mêmes plutôt que sur une norme instaurée par une quelconque autorité référente.

5.5.1 Analyse de la mesure d'évaluation WindowDiff

Avant de réaliser quelque étude que ce soit concernant la mesure *WindowDiff*, il est possible d'observer un premier biais (mineur certes) relatif à l'importance moindre accordée par la mesure aux frontières situées en tout début et fin de texte. Alors que ces frontières peuvent paraître aussi importantes que les autres (elles peuvent par exemple correspondre à la séparation des quelques phrases d'introduction, très générales, de la suite du texte pouvant être relatif à un point bien spécifique), la formule 5.3 (donnée en section 5.3.2) ne pénalise pas autant les erreurs situées entre les phrases 1 et k et entre les phrases $n-k+1$ et n autant que les autres (tel que défini dans la formule 5.3, k correspond à la taille de la fenêtre de calcul utilisée par la mesure). Puisque situées en bord de texte, ces positions sont considérées

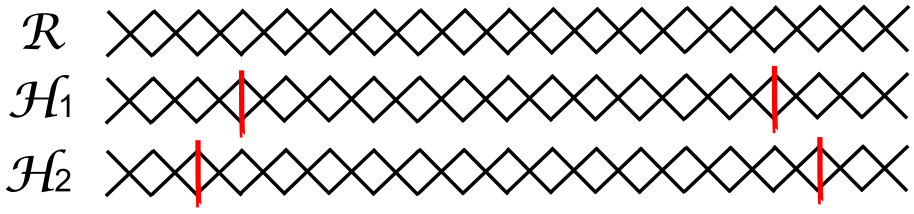

FIGURE 5.7 – Extrémités du texte et WindowDiff

un plus petit nombre de fois que celles situées entre les phrases k et $n-k+1$ par

la fenêtre de calcul de *WindowDiff*. La figure 5.7 illustre deux exemples de segmentation ($\mathcal{H}_1$ et $\mathcal{H}_2$) d'un texte de 18 phrases dont la référence ($\mathcal{R}$) ne possède aucune frontière (donc $k = 9$). Dans cette figure, les phrases sont représentées par des croix et les frontières par des barres horizontales. À première vue, les deux segmentations présentent un même taux d'erreur : toutes deux définissent deux frontières alors que la référence n'en possède pas. Elles devraient alors obtenir un score *WindowDiff* identique. Or, la segmentation $\mathcal{H}_2$, dont les erreurs sont situées plus en bord de texte, est effectivement moins pénalisée que $\mathcal{H}_1$: $\mathit{WindowDiff}(\mathcal{H}_2) = \frac{4}{9}$ alors que $\mathit{WindowDiff}(\mathcal{H}_1) = \frac{6}{9}$. Ce problème pourrait être réglé en ajoutant k phrases fictives en début et fin de texte et évaluer la segmentation en prenant en compte ces décalages dans les calculs. Alternativement, il peut être résolu en transformant la formule 5.3 de la manière suivante :

$$\mathit{WindowDiff}(\mathcal{H}, \mathcal{R}) = \frac{1}{(n+k-2)} \times \left(\begin{array}{ll} \sum_{i=2}^{k} (|fr(\mathcal{R}, 1, i) - fr(\mathcal{H}, 1, i)|) & + \\ \sum_{i=1}^{n-k} (|fr(\mathcal{R}, i, i+k) - fr(\mathcal{H}, i, i+k)|) & + \\ \sum_{i=n-k+1}^{n-1} (|fr(\mathcal{R}, i, n) - fr(\mathcal{H}, i, n)|) & \end{array} \right) \tag{5.18}$$

Considérant cette nouvelle formule, nous cherchons maintenant à étudier l'influence que peuvent avoir le nombre de frontières déterminées dans la segmentation à évaluer et le nombre de frontières présentes dans la segmentation de référence sur le score de *WindowDiff* comparant ces deux segmentations. Afin d'estimer la probabilité d'obtenir un score élevé de *WindowDiff* selon ces deux nombres de frontières, 1000 instances de $\mathcal{B}_\mathcal{R}$ et $\mathcal{B}_\mathcal{H}$ ont été générées aléatoirement, sur un texte fictif de 100 phrases ($b_i \in \{1, \ldots, 99\}$), pour chaque couple de nombres de frontières $(|\mathcal{B}_\mathcal{R}|, |\mathcal{B}_\mathcal{H}|) \in \{0, \ldots, 20\}^2$ possibles.

Les graphes de la figure 5.8 représentent les scores de *WindowDiff* obtenus sur ces différentes segmentations générées aléatoirement. Chaque courbe donne l'évolution de la moyenne des scores obtenus avec un nombre de frontières de la référence $|\mathcal{B}_\mathcal{R}|$ donné (1, 5, 10 ou 15) selon le nombre de frontières de la segmentation à évaluer $|\mathcal{B}_\mathcal{H}|$ (chaque point correspond à la moyenne de 1000 comparaisons de deux segmentations $\mathcal{B}_\mathcal{R}$ et $\mathcal{B}_\mathcal{H}$). Le fait que les courbes obtenues ne soient pas des droites horizontales montre que l'espérance mathématique du score obtenu par la mesure *WindowDiff* est influencée par le nombre de frontières déterminées. D'après les quatre courbes présentées, le risque d'obtenir un taux élevé d'erreur selon *WindowDiff* augmente avec le nombre de frontières que la méthode de segmentation détermine (d'autant plus sur les textes ne possédant que peu de frontières à retrouver). La mesure risque alors de défavoriser les méthodes très sensibles aux ruptures thématiques des textes. Alors que les mesures de rappel et de précision s'équilibrent l'une l'autre (lorsque

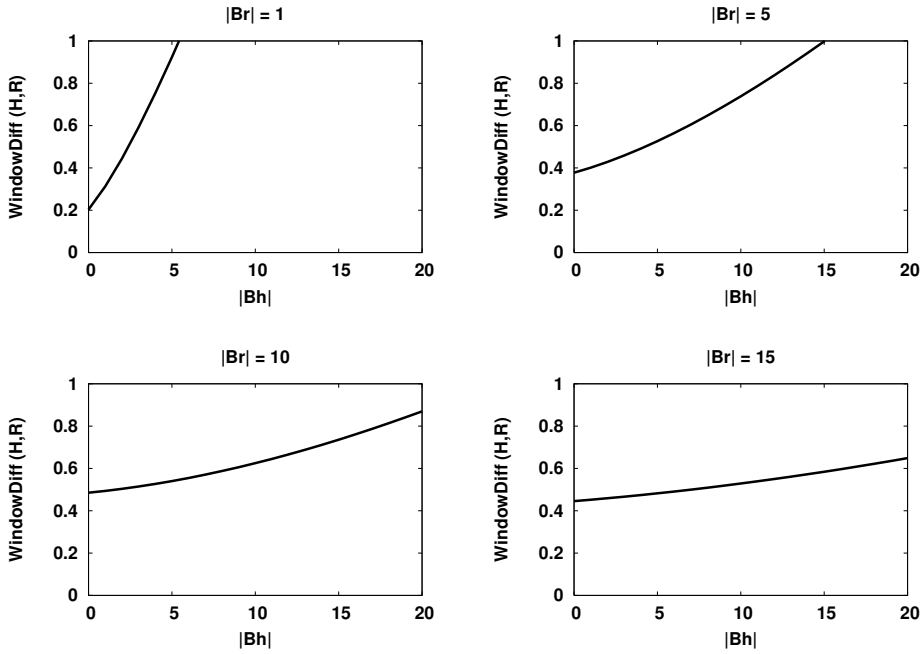

FIGURE 5.8 – Tendances de WindowDiff selon le nombre de frontières déterminées

le nombre de frontières déterminées augmente, l'espérance de la précision diminue d'autant que l'espérance du rappel augmente), rien ne vient contre-balancer les variations d'espérance de la mesure *WindowDiff*. Cela pose problème pour la comparaison des méthodes : par exemple, si une méthode a déterminé 10 frontières et une autre aucune sur un même texte possédant 5 frontières à retrouver, la segmentation ne présentant aucune frontière a bien plus de chances d'être jugée de meilleure qualité que celle en présentant 10. En fait, pour que la segmentation déterminant 10 frontières puisse égaler le score de la segmentation n'en déterminant aucune, il faudrait que 5 de ses frontières correspondent aux 5 frontières de la référence. Dans ce cas, elle pourrait être considérée comme très performante puisqu'elle a retrouvé l'ensemble des frontières de la référence (son rappel est alors égal à 1), les frontières supplémentaires n'étant que le résultat d'une plus grande sensibilité aux ruptures thématiques (ces frontières additionnelles peuvent peut-être se justifier d'une manière ou d'une autre). Néanmoins, même dans ce cas idéal, cette segmentation n'obtiendrait qu'un score identique à celle ne présentant aucune frontière. Il apparaît alors clairement que, au delà des problèmes que pose la rigidité inhérente à la comparaison d'une segmentation obtenue avec une segmentation de référence, la mesure *WindowDiff* ne permet pas d'évaluer les méthodes de manière équitable. Tel qu'énoncé en section 5.4.3, l'utilisation de cette mesure n'est alors valide que si les deux méthodes comparées ont tendance à déterminer des nombres de frontières

proches.

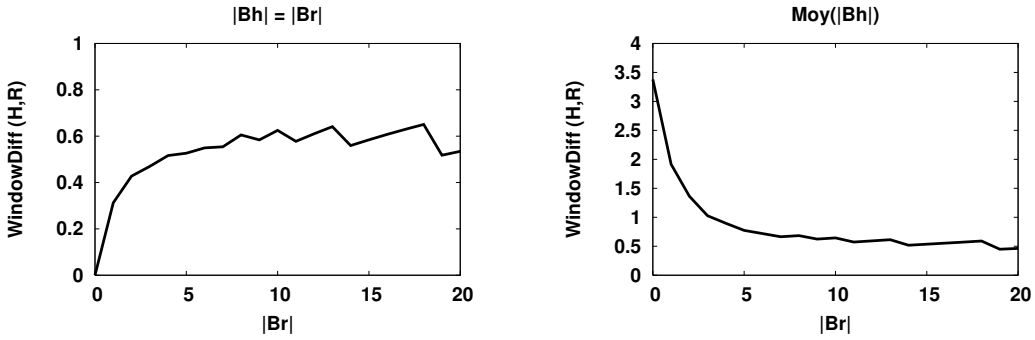

FIGURE 5.9 – Tendances de WindowDiff selon le nombre de frontières de référence

Par ailleurs, il semble que l'espérance de la mesure soit également influencée par le nombre de frontières à retrouver dans la segmentation de référence. La figure 5.9 souligne ce phénomène : son graphe de gauche, qui représente la courbe des scores obtenus avec des nombres de frontières égaux dans les deux segmentations (chaque point correspondant à une moyenne de 1000 comparaisons), montre que, même si le nombre de frontières de la segmentation à évaluer correspond parfaitement au nombre de frontières de référence, l'espérance de WindowDiff varie selon le nombre de frontières à retrouver dans le texte sur lequel porte l'évaluation [27]. Le graphe de droite, plus significatif, représente quant à lui la courbe des moyennes des scores obtenus pour chaque nombre de frontières de la segmentation $\mathcal{H}$ appartenant à $\{0, \ldots, 20\}$ selon le nombre de frontières de référence (chaque point correspond alors à 21000 comparaisons). Il montre clairement que plus il y a de frontières dans la segmentation de référence, plus les risques de pénalité diminuent. Ces variations d'espérance induisent des différences de poids entre les textes utilisés pour l'évaluation des systèmes, ce qui peut conduire à des conclusions biaisées. Étant donné que le risque de pénalité que l'on prend en déterminant un nombre trop important de frontières sur des textes n'en possédant que peu à retrouver est plus important que celui que l'on prend en ne déterminant pas assez de frontières dans des textes en possédant beaucoup, ce phénomène conduit, encore une fois, à favoriser les méthodes peu sensibles aux changements thématiques. Ce problème pourrait être corrigé en n'utilisant que des textes sur lesquels l'espérance de score *WindowDiff* est la même mais ce genre de corpus est difficile (voire impossible) à construire, le nombre de phrases que les textes contiennent ayant lui aussi un impact sur l'espérance du score de *WindowDiff*.

Afin de vérifier ces observations sur des segmentations réelles (en opposition

27. Les différentes vagues observables sur la courbe correspondent aux changements de la taille de la fenêtre utilisée, k étant dépendant du nombre de frontières contenues par la segmentation de référence.

avec les segmentations aléatoires jusqu'alors utilisées), nous comparons entre elles plusieurs versions de la méthode de segmentation *ClassStruggle*, dont le paramètre β permet d'influer directement sur le nombre de frontières obtenues. Connaissant les valeurs optimales pour ce paramètre (entre 0.7 et 0.75), la comparaison entre les versions de cette méthode nous permet d'observer les différences entre scores obtenus par la mesure *WindowDiff* et les performances réelles des systèmes. Le tableau 5.3 donne les résultats obtenus par ces différentes versions sur le corpus $AP(50, 4)$ utilisé précédemment (voir section 5.4.3), selon les mesures classiques de rappel, précision, nombre de frontières et donc la mesure *WindowDiff* qui nous intéresse. L'ensemble des versions de *ClassStruggle* utilisent le même jeu de clusters initiaux, issus de la méthode de clustering de *Nuées Dynamiques* employée pour obtenir les résultats présentés en section 5.4.3. Les résultats obtenus par d'autres méthodes sont reportés dans ce tableau à titre indicatif.

AP(50,4)	TT	$C99$	JI	S	$C_{0.6}$	$C_{0.65}$	$C_{0.7}$	$C_{0.75}$	$C_{0.8}$	$C_{0.85}$
WindowDiff	0.40	0.38	0.29	0.19	0.52	0.42	0.27	0.23	0.25	0.32
Precision	0.49	0.48	0.57	0.68	0.39	0.44	0.58	0.60	0.63	0.50
Rappel	0.49	0.59	0.64	0.60	0.64	0.68	0.67	0.60	0.52	0.44
Nb. Frontières	4.31	4.94	4.21	3.49	6.34	5.38	4.47	3.96	3.42	2.48

TABLE 5.3 – Résultats avec les mesures d'évaluation classiques

Les résultats présentés dans ce tableau semblent confirmer les observations faites sur des segmentations aléatoires : alors que, selon les mesures de rappel et de précision, la méthode $C_{0.7}$ semble être celle permettant l'obtention des meilleurs résultats, le critère de *WindowDiff* attribue le meilleur score à la version $C_{0.75}$ qui semble moins sensible aux ruptures thématiques. La mesure lui préfère même la version $C_{0.8}$ qui semble pourtant produire des segmentations de bien moins bonne qualité puisque le score obtenu sur le critère de rappel lui est inférieur de 0.15 points, pour un gain de précision de seulement 0.05 points. Enfin, le fait que la version $C_{0.85}$ obtienne un score de précision inférieur à celui des versions $C_{0.7}$, $C_{0.75}$ et $C_{0.8}$ alors que le nombre de frontières qu'elle détermine est bien plus faible dénote d'une perte évidente d'efficacité (puisque la précision devrait augmenter avec la diminution du nombre de frontières). Cette version obtient néanmoins, grâce au faible nombre de frontières qu'elle détermine, un score de *WindowDiff* relativement bon, bien meilleur en tout cas que celui obtenu par les versions $C_{0.6}$ et $C_{0.65}$ qui semblent pourtant produire des segmentations de moins mauvaise qualité.

Ces importantes contradictions entre résultats ne peuvent pas être entièrement justifiées par le seul fait que la mesure *WindowDiff* considère les décalages de frontières de manière moins pénalisante que les mesures de rappel et de précision (qui les considèrent comme des absences ou des insertions abusives de frontières).

Sa tendance à favoriser les méthodes peu sensibles aux changements thématiques est évidente. La section suivante est alors dédiée à l'établissement d'une fonction de normalisation des résultats obtenus par la mesure *WindowDiff*.

5.5.2 Prise en compte des risques encourus

Si nous disposions d'une fonction permettant de modéliser l'espérance mathématique $E(\mathcal{H}, \mathcal{R})$ des scores de la mesure *WindowDiff* selon les nombres de frontières des deux segmentations $\mathcal{H}$ et $\mathcal{R}$, il serait alors possible de normaliser les résultats selon les risques encourus par les différentes méthodes :

$$NWin(\mathcal{H}, \mathcal{R}) = \frac{WindowDiff\,(\mathcal{H}, \mathcal{R})}{E(\mathcal{H}, \mathcal{R})} \tag{5.19}$$

Par l'application d'une telle opération, l'espérance de la mesure serait identique (égale à 1) pour toutes les méthodes de segmentation, quelle que soit leur sensibilité face aux changements thématiques, et pour tous les textes, quel que soit le nombre de frontières que leur segmentation de référence contient. Cela permettrait alors de comparer les résultats des différentes méthodes de manière pus équitable. Nous nous employons donc maintenant à la recherche d'une telle fonction de normalisation.

Le nombre de segmentations possibles selon un nombre de frontières donné est égal à $C_{n-1}^{|\mathcal{B}_\mathcal{R}|}$ pour la segmentation de référence et $C_{n-1}^{|\mathcal{B}_\mathcal{H}|}$ pour la segmentation à évaluer. Avec k la taille de la fenêtre utilisée par la mesure *WindowDiff* et n le nombre de phrases du texte, les probabilités $P(X = i \mid \mathcal{B}_\mathcal{R})$ et $P(X = i \mid \mathcal{B}_\mathcal{H})$ d'avoir i frontières dans une fenêtre donnée sont alors, selon la segmentation considérée, égales à :

$$P(X = i \mid \mathcal{B}_\mathcal{R}) = \frac{C_k^i \times C_{n-1-k}^{|\mathcal{B}_\mathcal{R}|-i}}{C_{n-1}^{|\mathcal{B}_\mathcal{R}|}} \tag{5.20}$$

$$P(X = i \mid \mathcal{B}_\mathcal{H}) = \frac{C_k^i \times C_{n-1-k}^{|\mathcal{B}_\mathcal{H}|-i}}{C_{n-1}^{|\mathcal{B}_\mathcal{H}|}} \tag{5.21}$$

On peut alors calculer la probabilité $P(i \neq j)$ d'avoir, sur une portion de texte donnée, un nombre différent de frontières dans les deux segmentations :

$$P(i \neq j) = \sum_{i=0}^{\min(k,|\mathcal{B}_\mathcal{R}|)} \sum_{j=0, j\neq i}^{\min(k,|\mathcal{B}_\mathcal{H}|)} \frac{C_k^i \times C_{n-1-k}^{|\mathcal{B}_\mathcal{R}|-i}}{C_{n-1}^{|\mathcal{B}_\mathcal{R}|}} \times \frac{C_k^j \times C_{n-1-k}^{|\mathcal{B}_\mathcal{H}|-j}}{C_{n-1}^{|\mathcal{B}_\mathcal{H}|}} \tag{5.22}$$

où $\min(k, |\mathcal{B}_\mathcal{R}|)$ correspond au minimum entre la taille de fenêtre k et le nombre de frontières de la segmentation de référence $|\mathcal{B}_\mathcal{R}|$ (et $\min(k, |\mathcal{B}_\mathcal{H}|)$ le même minimum mais appliqué à la segmentation à évaluer). Le score de *WindowDiff* représente une moyenne de scores obtenus sur l'ensemble des positions de fenêtre possibles. Dans

la version initiale de la mesure (formule 5.3), la probabilité $P(i \neq j)$ étant la même pour l'ensemble des fenêtres, l'espérance mathématique du score de *WindowDiff* est alors :

$$E(\mathcal{H},\mathcal{R}) = \sum_{i=0}^{\min(k,|\mathcal{B}_\mathcal{R}|)} \sum_{j=0}^{\min(k,|\mathcal{B}_\mathcal{H}|)} \frac{C_k^i \times C_{n-1-k}^{|\mathcal{B}_\mathcal{R}|-i}}{C_{n-1}^{|\mathcal{B}_\mathcal{R}|}} \times \frac{C_k^j \times C_{n-1-k}^{|\mathcal{B}_\mathcal{H}|-j}}{C_{n-1}^{|\mathcal{B}_\mathcal{H}|}} \times |i-j| \quad (5.23)$$

Néanmoins, afin de considérer les erreurs commises aux extrémités du texte de la même manière que les autres, nous avons transformé la formule initiale de *WindowDiff* en ajoutant des fenêtres de tailles variables en début et fin de texte (formule 5.18). Ces fenêtres supplémentaires doivent alors être prises en considération dans le calcul de l'espérance réalisé :

$$E(\mathcal{H},\mathcal{R}) = \frac{1}{(n+k-2)} \times \left(\begin{array}{l} 2 \times \sum_{f=1}^{k-1} \left(\sum_{i=0}^{\min(f,|\mathcal{B}_\mathcal{R}|)} \sum_{j=0}^{\min(f,|\mathcal{B}_\mathcal{H}|)} \frac{C_f^i \times C_{n-1-f}^{|\mathcal{B}_\mathcal{R}|-i}}{C_{n-1}^{|\mathcal{B}_\mathcal{R}|}} \times \frac{C_f^j \times C_{n-1-f}^{|\mathcal{B}_\mathcal{H}|-j}}{C_{n-1}^{|\mathcal{B}_\mathcal{H}|}} \times |i-j| \right) + \\ (n-k) \times \left(\sum_{i=0}^{\min(k,|\mathcal{B}_\mathcal{R}|)} \sum_{j=0}^{\min(k,|\mathcal{B}_\mathcal{H}|)} \frac{C_k^i \times C_{n-1-k}^{|\mathcal{B}_\mathcal{R}|-i}}{C_{n-1}^{|\mathcal{B}_\mathcal{R}|}} \times \frac{C_k^j \times C_{n-1-k}^{|\mathcal{B}_\mathcal{H}|-j}}{C_{n-1}^{|\mathcal{B}_\mathcal{H}|}} \times |i-j| \right) + \end{array} \right) \quad (5.24)$$

La figure 5.10 donne, à la manière de la figure 5.7 précédemment présentée, deux exemples de segmentation ($\mathcal{H}_1$ et $\mathcal{H}_2$) d'un texte composé de 18 phrases. En comparant les frontières contenues dans ces deux segmentations avec celles de la référence ($\mathcal{R}$), il paraît clair que $\mathcal{H}_2$ respecte mieux que $\mathcal{H}_1$ les ruptures thématiques identifiées lors de la constitution de la segmentation de référence : alors que $\mathcal{H}_1$ contient une unique frontière à une distance de 4 phrases de la frontière de référence, $\mathcal{H}_2$ possède l'une de ses deux frontières placée exactement à la même position que la frontière de référence. Or, $\mathcal{H}_1$ obtient un meilleur score de *WindowDiff* que $\mathcal{H}_2$: $\mathit{WndowDiff}(\mathcal{H}_1) = \frac{8}{25}$ alors que $\mathit{WndowDiff}(\mathcal{H}_2) = \frac{10}{25}$. La frontière additionnelle de

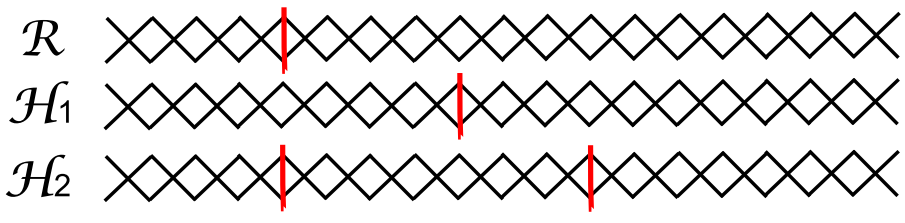

FIGURE 5.10 – Comparaison de segmentations aux nombres de frontières différents

$\mathcal{H}_2$, qui peut-être se justifie par une rupture qui avait été jugée trop faible lors de la constitution de la référence, pénalise très fortement le score obtenu par la segmentation. La normalisation de la mesure permet de corriger le problème en mettant en avant la supériorité de $\mathcal{H}_2$: $NWin(\mathcal{H}_2) = 0.906$ alors que $NWin(\mathcal{H}_1) = 1.023$. Certes, une segmentation qui n'aurait pas contenu de frontière supplémentaire aurait obtenu un meilleur score que $\mathcal{H}_2$, mais la normalisation établie dans cette section permet de limiter les biais liés aux différences de sensibilité face aux changements thématiques.

Le score *NWin* optimal qu'une segmentation peut obtenir selon le nombre de frontières qu'elle possède, c'est à dire le score obtenu si toutes les frontières découvertes ont une correspondance dans la référence (Precision=1) dans le cas de solutions en sous-segmentation ou le score obtenu si toutes les frontières de la référence ont été identifiées (Rappel=1) dans le cas de solutions en sur-segmentation, correspond à :

$$O(\mathcal{H}, \mathcal{R}) = \frac{k \times \mid |\mathcal{B}_{\mathcal{R}}| - |\mathcal{B}_{\mathcal{H}}| \mid}{(n + k - 2) \times E(\mathcal{H}, \mathcal{R})} \tag{5.25}$$

Si l'absence ou l'insertion de frontières par rapport à la référence pouvait se justifier d'une certaine manière, il serait possible de soustraire ce score optimal du score du *WindowDiff* normalisé pour obtenir un score dépendant uniquement des frontières comparables entre segmentation à évaluer et segmentation de référence. La segmentation serait alors évaluée sur les positions de ses frontières trouvant une correspondance dans la référence. Néanmoins, un nombre trop important (ou trop faible) de frontières n'est pas nécessairement le résultat d'une sensibilité différente face aux ruptures thématiques. Il peut tout aussi bien s'agir d'erreurs de la part de la méthode évaluée. Par la comparaison d'une segmentation avec une référence, il n'est pas possible de tirer de conclusion concernant la justification de ces frontières manquantes ou additionnelles. Par conséquent, afin d'obtenir une mesure qui soit plus flexible par rapport à la référence établie, nous considérons que la justification ou non de telles frontières sont des événements équiprobables et nous proposons alors une mesure plus tolérante, qui soustrait du score normalisé obtenu par une segmentation le score optimal qu'elle aurait pu obtenir (selon le nombre de frontières qu'elle possède) multiplié par un coefficient de tolérance fixé à $\frac{1}{2}$:

$$TNWin(\mathcal{H}, \mathcal{R}) = NWin(\mathcal{H}, \mathcal{R}) - \frac{1}{2} \times O(\mathcal{H}, \mathcal{R}) \tag{5.26}$$

Le tableau 5.2 présente les scores du *WindowDiff* normalisé ($NWin$) et du *WindowDiff* normalisé tolérant ($TNWin$) obtenus par nos différentes méthodes sur le même corpus $AP(50, 4)$ qu'utilisé précédemment.

Au vu des résultats reportés ici, l'application de notre fonction de normalisation paraît limiter efficacement les biais présentés par la mesure d'évaluation *WindowDiff*. En effet, le classement entre méthodes qu'il est possible d'établir selon les

AP(50,4)	TT	$C99$	JI	S	$C_{0.6}$	$C_{0.65}$	$C_{0.7}$	$C_{0.75}$	$C_{0.8}$	$C_{0.85}$
NWin	0.71	0.65	0.53	0.39	0.73	0.66	0.49	0.46	0.55	0.79
TNWin	0.65	0.58	0.47	0.31	0.62	0.56	0.41	0.43	0.49	0.67

TABLE 5.4 – Résultats avec les mesures normalisées

scores de $NWin$ semble bien plus cohérent que celui que l'on pouvait établir dans la section précédente selon la mesure *WindowDiff*[28]. Cette mesure permet alors une meilleure appréciation des performances réelles des méthodes. Néanmoins, malgré la normalisation opérée, les différences de sensibilité continuent de pénaliser certaines méthodes, la base de la mesure restant une comparaison avec une segmentation de référence. Par exemple, le nombre trop important de frontières déterminées par la méthode $C_{0.7}$ ne lui permet pas de démontrer sa supériorité par rapport à $C_{0.75}$. La tolérance accrue de la mesure $TNWin$ par rapport aux différences de sensibilité des méthodes permet de corriger ce problème. Le fait de soustraire le score optimal que la segmentation aurait pu obtenir permet de la juger uniquement sur les frontières trouvant une correspondance dans la référence. Cependant, la détermination arbitraire du coefficient de tolérance (qui correspond à un degré de confiance que l'on peut attribuer aux frontières ne trouvant pas de correspondance) laisse planer une certaine incertitude sur l'évaluation. Des tests sur des corpus autres que celui employé pour l'obtention des résultats présentés ici nous laissent néanmoins penser que cette prise de distance par rapport à la segmentation de référence permet une meilleure estimation des performances relatives des méthodes.

5.5.3 Respect des différences

Jugeant difficile de tirer davantage de tolérance d'une évaluation basée sur des comparaisons entre segmentations à évaluer et segmentations de référence (et la constitution de références posant en outre de nombreuses difficultés), nous proposons ici d'établir un processus d'évaluation centré sur les méthodes de segmentation elles mêmes, afin de les évaluer sur des caractéristiques qui leur sont propres plutôt que de comparer leurs sorties à des références définies de manière fixe.

Tel que mentionné à de maintes reprises, les segments thématiques d'un texte peuvent être considérés comme des parties de texte présentant à la fois une forte cohésion des éléments qui les composent et une grande dissimilarité avec les éléments qui les entourent [Salton *et al.*, 1996]. La cohésion interne des segments et la dissimilarité entre segments adjacents sont alors deux critères déterminant de la qualité d'une segmentation. Or, nous notons que l'ordre entre phrases de même segment

28. Il paraît en effet bien plus en accord avec le classement que l'on peut réaliser en effectuant une étude approfondie des résultats obtenus selon les mesures existantes.

n'agit sur aucun de ces deux critères : la modification de l'ordre des phrases qu'un segment contient ne change ni le fait que la partie de texte concernée aborde une même thématique, ni le fait que les segments qui l'entourent en abordent une autre bien différente. Cette observation nous laisse envisager trois hypothèses qui sont à la base de la méthodologie d'évaluation que nous avons mise en place :

- **Hypothèse 1 :** Des permutations entre phrases d'un même segment ne devraient pas induire de modification dans la détection des frontières si les segments initiaux sont bien formés (et si cela n'est pas dû au hasard). C'est à dire qu'une nouvelle application de la méthode sur le texte ainsi modifié devrait conduire à l'obtention des mêmes frontières que celles déterminées initialement.
- **Hypothèse 2 :** Si les frontières déterminées par la méthode de segmentation à évaluer sont incorrectes (*i.e.,* elles ne peuvent pas se justifier d'une quelconque manière), le fait de modifier l'ordre des phrases appartenant à un même segment conduit à l'obtention de frontières différentes de celles déterminées initialement [29].
- **Hypothèse 3 :** Moins une méthode de segmentation est efficace, moins elle est stable (*i.e.,* moins elle a de chances de retrouver ses frontières initiales) face à des permutations entre phrases de même segment.

Si ces hypothèses pouvaient se vérifier, un processus d'évaluation tel que celui décrit par l'algorithme 5.2 pourrait être envisagé. Il s'agit d'estimer la capacité d'une méthode de segmentation à retrouver les frontières qu'elle avait elle-même initialement déterminées sur le texte concerné, et ce malgré des permutations opérées entre phrases de même segment initial. Les frontières initialement déterminées par la méthode peuvent alors être considérées comme la segmentation de référence à laquelle les différentes segmentations du texte modifié se comparent. Le score de stabilité obtenu correspond à une *F1-mesure* (formule 3.4 avec le coefficient $\beta = 1$), dont le rappel correspond à la proportion de frontières initiales retrouvées dans la nouvelle segmentation et la précision à la proportion de nouvelles frontières correspondant à des frontières de la segmentation initiale. Plus le score obtenu est élevé, plus la méthode est stable face aux modifications du texte et plus elle peut être considérée comme performante.

Afin de valider ce processus d'évaluation, nous procédons maintenant à la vérification des hypothèses sur lesquelles il repose.

Hypothèse 1

L'étude réalisée ici porte sur deux groupes D_1 et D_2 de 20 documents issus du corpus $AP(50, 4)$. Les documents utilisés résultant de la concaténation de cinq articles chacun, nous avons tout d'abord demandé à deux groupes G_1 et G_2 de 10 personnes volontaires de rechercher les séparations entre articles contenues par les

29. Cela peut en effet permettre de faire s'éloigner des phrases d'incises ayant conduit au découpage d'une partie de texte traitant d'une même thématique, ou des phrases de transition ayant caché l'existence d'une rupture thématique.

Algorithme 5.2 : Algorithme du Test de Stabilité

Données :
Une liste d'unités $\mathcal{U}$,
Une méthode de segmentation à évaluer,
Un nombre Max d'itérations.
Résultat :
Le score TS du test de stabilité.

début
  $\mathcal{B}_\mathcal{R} = \{\}$; $\mathcal{B}_\mathcal{H}{}' = \{\}$; $\mathcal{R} = (\mathcal{U}, \{\})$; $\mathcal{H} = (\mathcal{U}, \{\})$; $\mathcal{U}' = \mathcal{U}$;
  $Rap = 0$; $Prec = 0$; $Nb = 0$; $TS = 0$;
  $\mathcal{B}_\mathcal{R}$ =Segmentation du texte (liste $\mathcal{U}$) avec la méthode à évaluer;
  $\mathcal{R} = (\mathcal{U}, \mathcal{B}_\mathcal{R})$;
  tant que $Nb < Max$ **faire**
    $\mathcal{U}'$ =Permutations aléatoires, dans $\mathcal{U}$, entre unités
      de même segment selon la segmentation $\mathcal{R}$;
    $\mathcal{B}_\mathcal{H}$ =Segmentation du texte (liste $\mathcal{U}'$) avec la méthode à évaluer;
    $\mathcal{H} = (\mathcal{U}, \mathcal{B}_\mathcal{H})$;
    $Rap = Rap + Rappel(\mathcal{H}, \mathcal{R})$;
    $Prec = Prec + Precision(\mathcal{H}, \mathcal{R})$;
    $Nb = Nb + 1$;
  fin
  $Rap = Rap/Nb$; $Prec = Prec/Nb$;
  $TS = F1(Rap, Prec)$;
fin

différents documents, les sujets du groupe G_1 travaillant sur les documents de D_1 et les sujets de G_2 travaillant sur les documents de D_2. En comparant les résultats obtenus avec les véritables séparations entre articles, nous nous sommes aperçus que personne n'avait fait d'erreur [30]. Il est alors possible de considérer que les personnes impliquées dans l'étude constituent des méthodes de segmentation parfaites (qui ne commettent jamais d'erreur).

L'objectif de l'étude étant de vérifier l'hypothèse stipulant que, si les frontières déterminées sont correctes (*i.e.*, correspondent à des changements thématiques effectifs), et que cela n'est pas simplement dû à un coup de chance, des permutations entre phrases de même segment ne risquent pas de perturber la détection des frontières. Nous avons alors permuté aléatoirement les phrases de même article et présenté les documents ainsi obtenus aux sujets (pour un même texte, les permutations effectuées sont différentes pour chacun sujet). Afin de ne pas biaiser l'étude, nous avons in-

30. Ces résultats sont bien meilleurs que ceux reportés dans l'étude réalisée par Hearst (que l'on mentionne en section 5.3.1), qui avait montré de très nombreuses divergences entre les segmentations proposées. Cela est dû au fait qu'ici, nous travaillons avec des articles concaténés, ce qui conduit à une détermination des frontières largement facilitée.

terverti l'affectation du groupe de sujets au groupe de documents étudiés[31] : les sujets de G_1 travaillent maintenant sur les documents de D_2 et les sujets de G_2 sur les documents de D_1. Une fois les différentes segmentations réalisées, nous comparons les frontières obtenues avec les frontières initiales (c'est à dire les séparations entre articles) : encore une fois, personne n'a fait d'erreur. L'hypothèse 1 semble alors valide, des permutations entre phrases de même segment ne perturbent pas la détection des frontières si les frontières initiales correspondent à des transitions thématiques effectives.

Hypothèse 2

La vérification de la validité de l'hypothèse 2 s'est faite de manière similaire : deux groupes de 10 sujets doivent repérer des séparations entre articles de deux jeux de documents D_1 et D_2. Afin de simuler des méthodes de segmentation imparfaites, plutôt que d'utiliser des articles en anglais pour constituer les documents de l'étude, chaque document de D_1 et D_2 est le résultat de la concaténation de cinq articles du journal allemand "Die Welt". Aucun des sujets impliqués dans l'étude ne parlant allemand, les résultats obtenus sont bien moins bons que ceux obtenus avec des documents du corpus $AP(50, 4)$: seules 10% des frontières détectées par les individus se sont avérées correspondre à des séparations entre articles. Les individus de l'étude peuvent alors maintenant simuler des méthodes de segmentation très imparfaites.

Le fait de permuter les phrases à l'intérieur des différents segments obtenus "détruit" la structure du document, les articles qui le composent étant mélangés. C'est alors sans surprise que nous avons observé des frontières bien différentes de celles initialement découvertes. La deuxième hypothèse semble alors se vérifier, des permutations entre phrases de même segment initial conduisent à des perturbations dans la détection des frontières lorsque les frontières initiales ne correspondent pas à des ruptures thématiques effectives.

Hypothèse 3

Les degrés de performances des sujets étant difficiles à estimer, nous procédons ici à la vérification de la troisième hypothèse par application du test de stabilité sur les méthodes de segmentation utilisées pour vérifier l'efficacité des mesures de normalisation de *WindowDiff*. Le tableau 5.5 présente alors les résultats obtenus sur le corpus $AP(50, 4)$[32]. Afin de vérifier la troisième hypothèse, l'objectif est ici d'obtenir le même classement des méthodes (selon leur score de stabilité) qu'il est possible d'établir à partir de l'analyse conjointe des mesures de rappel et de précision

31. Si l'on avait présenté les mêmes documents au même groupe de sujets, l'étude aurait pu être biaisée par le fait que les personnes se souviennent des différents articles constituant chacun des documents.

32. Le test de stabilité ne peut pas s'appliquer à la méthode *SegGen* qui n'est pas une méthode déterministe. C'est la raison pour laquelle cette méthode ne figure pas dans le tableau 5.5.

présentées dans le tableau 5.3. Le nombre d'itérations [33] (nombre de segmentations successives du texte modifié selon les segments initiaux) réalisées par le test de stabilité pour obtenir les résultats présentés est de 100.

AP(50,4)	TT	$C99$	JI	$C_{0.6}$	$C_{0.65}$	$C_{0.7}$	$C_{0.75}$	$C_{0.8}$	$C_{0.85}$
F_1	0.61	0.66	0.74	0.63	0.69	0.76	0.74	0.70	0.59

TABLE 5.5 – Résultats du test de Stabilité

Les résultats présentés sont sensiblement les mêmes que ceux obtenus par la mesure *TNWin* qui semblait, dans la section précédente, permettre le meilleur classement des méthodes. La troisième hypothèse semble donc se vérifier : plus une segmentation est performante, moins les permutations opérées entre phrases de même segment ne perturbent les calculs, plus le degré de stabilité observé est important.

La mesure que nous proposons ici semble permettre une évaluation des méthodes de segmentation efficace. Basée sur des applications successives d'une méthode sur un texte donné, elle ne peut cependant pas être appliquée aux méthodes de segmentation non déterministes. Elle ne peut pas être utilisée non plus sur des méthodes utilisant des marqueurs linguistiques de transition thématique puisque les permutations entre phrases risquent de placer ces marqueurs en milieu de segment et ainsi perturber les résultats. Néanmoins, en ce qui concerne la comparaison de méthodes statistiques déterministes, cette mesure présente quatre avantages majeurs :

- Elle permet de se dispenser de la tâche très difficile de constitution de segmentations de référence ;
- Puisque basée sur des comparaisons des frontières obtenues par une même méthode, elle permet d'évaluer les méthodes sur leurs propres caractéristiques, sans pénaliser une méthode ou une autre selon leur degré de sensibilité ou leur point de vue spécifique adopté ;
- Le fait de relancer de nombreuses fois la méthode à évaluer réduit les possibles coups de chance dont elle peut bénéficier. La taille du corpus d'évaluation peut alors être réduit : nos expérimentations portent sur un corpus de 350 textes mais le score obtenu par chaque méthode n'a que très peu varié après l'examen des trente premiers (contrairement aux autres critères qui nécessitent un nombre de textes d'évaluation bien plus important) ;
- Un degré de confiance peut être associé à la segmentation qu'une méthode produit.

33. Plus ce nombre est élevé, plus le degré de fiabilité des résultats obtenus l'est aussi. 100 itérations nous semble un nombre suffisant pour être à même de comparer efficacement les performances des systèmes.

5.6 Conclusion

La segmentation thématique des textes, qui trouve de nombreuses applications en recherche d'information, se révèle une tâche très complexe, tant par la mise en place de méthodes performantes que pour l'évaluation des méthodes proposées. La structuration d'un texte, et par là son découpage en segments thématiques, dépend de l'ensemble des éléments qui le constituent. C'est pourquoi, offrant une alternative à la plupart des approches existantes dans lesquelles la détermination des frontières se fait de manière isolée, nous avons proposé deux nouvelles méthodes de segmentation qui cherchent à adopter une vision plus globale des textes que la plupart des méthodes actuelles.

La première, la méthode *ClassStruggle*, utilise un clustering préliminaire des phrases du texte pour en faire émerger les thématiques principales. Dans cette méthode, la décision d'affecter une phrase à un segment donné est prise en considérant l'ensemble des autres affectations préalablement réalisées, ce qui permet d'obtenir une segmentation plus cohérente que si la détermination se faisait, tel que réalisé par la plupart des approches existantes, de manière locale. Par ailleurs, la distance prise par rapport au clustering initial permet de repérer uniquement les principales ruptures thématiques du document, en évitant de tomber dans le piège de segmenter le texte à chaque petite irrégularité du texte. Les résultats obtenus s'avèrent très compétitifs, notre méthode *ClassStruggle* présentant de meilleurs résultats que les méthodes existantes sur la plupart des corpus expérimentés. Dans de prochaines études, il pourra être intéressant d'évaluer dans quelle mesure l'utilisation de méthodes de clustering plus performantes peut avoir des effets bénéfiques sur la segmentation obtenue. La comparaison entre segmentations résultant de trois différentes méthodes de clustering semble indiquer que l'utilisation de techniques de clustering plus perfectionnées peuvent permettre des améliorations significatives des résultats. Cette approche nous apparaît alors très prometteuse.

Néanmoins, son aspect incrémental réduit sa capacité à optimiser deux critères importants : la cohésion interne de segments et la dissimilarité entre segments adjacents. Considérant la segmentation comme un problème d'optimisation multi-objectifs, nous avons proposé une autre approche, la méthode *SegGen*, qui utilise un algorithme génétique pour optimiser conjointement ces deux critères. Plutôt que de déterminer les frontières les unes après les autres, cette méthode oriente sa recherche par la considération de segmentations réalisées sur l'ensemble du texte. Cela lui permet d'adopter une vision complète des segments créés et d'optimiser ainsi la segmentation de manière globale. Malgré la simplicité des opérateurs génétiques utilisés, les expériences ont montré que cette approche réalise des segmentations de texte de bien meilleure qualité que celles obtenues par les algorithmes existants. Le fait de considérer le texte dans sa globalité nous paraît alors déterminant. Par ailleurs, le fait que *SegGen* obtienne de meilleurs résultats que la méthode proposée dans [Ji and Zha, 2003], qui considère elle aussi les textes globalement mais n'optimise

qu'un unique critère de cohésion interne des segments, montre que la dissimilarité entre segments adjacents s'avère être un critère important pour la segmentation d'un texte. De nombreuses améliorations peuvent être apportées à notre méthode, notamment par la recherche d'opérateurs plus évolués, ou par l'application de certains pré-traitements sur la matrice de similarités initiale. Par ailleurs, les ruptures thématiques d'un texte ne sont pas toujours tout à fait franches du fait de phrases de transition. Une étude approfondie des individus contenus par l'archive externe de l'algorithme peut amener à déterminer des intervalles de transition plutôt que de trancher de manière brutale. Enfin, un objectif supplémentaire pourrait permettre de réguler la longueur des segments selon les préférences de l'utilisateur.

L'aspect subjectif de la segmentation thématique induit de grandes difficultés pour la mise en place de mesures permettant une comparaison équitable des méthodes. Lors de la mise en place de nos deux méthodes de segmentation, nous nous sommes aperçus que la mesure *WindowDiff*, utilisée dans la plupart des études, présentait une forte tendance à favoriser les méthodes peu sensibles aux ruptures thématiques. Nous avons alors tout d'abord proposé une fonction de normalisation qui permet de réduire les biais de cette mesure d'évaluation et de prendre une certaine distance par rapport à la segmentation de référence à laquelle la segmentation à évaluer est comparée. Enfin, nous avons proposé une méthodologie d'évaluation totalement innovante, qui s'appuie sur une estimation du degré de stabilité de la méthode de segmentation. Cela permet d'éviter la lourde tâche de constitution de segmentations de référence, et surtout, de permettre aux segmentations d'être évaluées sur des caractéristiques qui leur sont propres plutôt que de se comparer à des frontières établies de manière fixe. Ces nouvelles mesures d'évaluation nous ont permis d'observer encore une fois les bonnes performances des deux méthodes de segmentation que nous avons proposées. Une participation à une campagne d'évaluation de la segmentation telle que celle proposée par la conférence DEFT'06 [Azé *et al.*, 2006] pourrait être utile pour confirmer ces observations mais les résultats expérimentaux que nous avons obtenus sur les corpus de *TREC* paraissent très prometteurs. Dans la suite de ce rapport, les segments thématiques manipulés sont les segments issus du processus de segmentation *SegGen* que nous avons proposé dans ce chapitre.

Chapitre 6

Segmenter pour ordonner les documents

Il a été montré à maintes reprises, dans le cadre des approches dites de *Passage Retrieval*, que l'individualisation des passages d'un document pouvait permettre d'améliorer la qualité de l'estimation de son degré de pertinence. Il semblerait naturel que les passages considérés correspondent aux segments résultant d'un processus de segmentation thématique tel que décrit au chapitre précédent. Néanmoins, ces passages semblent souffrir de la grande diversité de tailles qu'ils présentent, les mesures d'estimation de pertinence attribuant généralement des scores fortement dépendant du nombre de termes contenus par les textes considérés. Dans un premier temps, nous cherchons à mettre en place une normalisation de ces scores afin de redonner leur chance à ces passages de taille variable que sont les segments thématiques. Ce chapitre conclut alors sur des expérimentations visant à définir le type de passage le plus à même de servir les intérêts d'une recherche d'information.

Sommaire

6.1 Les approches de type *Passage Retrieval* 150
6.2 Mesures de pertinence et longueur des textes 152
6.2.1 Impacts de la longueur des textes sur les estimations de pertinence 157
6.2.2 Normalisation des scores de pertinence par régression statistique 160
6.3 Types de segments et performances des systèmes 164
6.3.1 Types de segments étudiés 165
6.3.2 Comparaison des approches 166
6.4 Conclusion 169

6.1 Les approches de type *Passage Retrieval*

Dans la plupart des systèmes de recherche d'information, la mise en correspondance d'un document avec une requête est réalisée de manière globale, en prenant en compte la totalité de son texte dans les calculs de similarité effectués. Néanmoins, tel que nous l'avons vu au chapitre précédent, les documents n'abordent pas nécessairement une unique thématique tout au long de leur discours. Ils peuvent en effet présenter de nombreux passages aux thématiques diverses. Un document peut alors très bien se trouver en totale déconnection avec le besoin de l'utilisateur dans une large partie de son discours, tout en possédant un petit extrait de texte qui contienne exactement les informations concernées par la recherche. Du fait que cet extrait soit noyé dans une masse d'informations sans rapport direct avec la requête formulée, le document le contenant risque de ne pas être retourné, ou alors d'être très mal positionné dans la liste de résultats présentée à l'utilisateur.

Des approches de recherche d'information alternatives, les approches dites de *Passage Retrieval* [Callan, 1994], visent à redonner leur chance à de tels documents (ou morceaux de documents). Dans ces approches, les documents sont considérés comme des ensembles de passages, qui peuvent alors être pris en compte de manière individuelle par les mesures de similarité. Outre le fait que cette individualisation des passages puisse permettre de présenter des fragments de texte souvent mieux ciblés et interprétables que de longs documents pouvant aborder parfois de nombreuses thématiques différentes, cette considération disjointe des passages permet de réaliser une estimation plus fine des potentiels de pertinence, ce qui conduit bien souvent à un meilleur classement des résultats retournés à l'utilisateur [1]. Le fait de se focaliser sur des zones de texte correspondant chacune à une thématique particulière peut en effet permettre d'obtenir des représentations plus précises des différents concepts abordés. De plus, ce découpage des documents permet de réintroduire une notion de localité des termes dans les représentations [2] utilisées par les mesures de similarité [Choquette, 1996]. En effet, pour que deux termes de la requête puissent être considérés ensemble (dans un même calcul de similarité), il leur faut être suffisamment proches l'un de l'autre pour appartenir au même passage, ce qui réduit alors le risque de retourner des documents dans lesquels les termes de la requête sont bien présents mais employés dans des contextes distincts (alors que l'association de ces termes dans la requête est supposée avoir été réfléchie pour définir un concept précis). Enfin, cela permet de privilégier les documents qui concentrent les occur-

1. Lorsque ces résultats sont des documents, le classement se fait généralement selon le score de leur passage le plus proche de la requête [Callan, 1994]. C'est l'utilisation la plus naturelle des scores des passages puisque cela correspond à ne s'intéresser qu'au passage potentiellement pertinent, le degré de similarité des passages aux thématiques déconnectées de la requête n'ayant que peu d'importance. Notons néanmoins que dans [Hearst and Plaunt, 1993], des expériences ont montré que de bons résultats pouvaient également être obtenus en réalisant, pour chaque document, la somme des scores obtenus par plusieurs de ses passages les plus proches de la requête.

2. La plupart des modèles de représentation, tel que le modèle vectoriel, ne considérant pas les positions relatives des termes dans les documents.

rences des termes de la requête dans une zone donnée de leur texte, et sont alors supposés faire une description relativement précise du sujet, par rapport à ceux qui n'en font mention que ponctuellement, et de façon uniforme tout au long de leur discours [Kaszkiel and Zobel, 2001].

De nombreuses expérimentations ont montré que la considération des passages de chaque document permettait effectivement d'obtenir des listes de résultats clairement mieux ordonnées que lorsque les documents sont considérés globalement [Hearst and Plaunt, 1993; Callan, 1994; Salton *et al.*, 1993] [3]. De nombreux types de passages ont alors été expérimentés :

- Passages issus de la structuration physique du texte (paragraphes, sections, etc...) : ce sont les passages les plus simples à obtenir mais, tel que mentionné au chapitre précédent, ils n'existent pas toujours dans les documents et, lorsqu'ils existent, ne correspondent pas toujours à la structuration réelle du discours [Callan, 1994]. Les résultats obtenus avec ce type de passages ne sont pas toujours probants [Kaszkiel and Zobel, 2001] ;
- Segments thématiques : tels que présentés en détail dans le chapitre précédent, ces passages sont issus d'un processus de segmentation automatique qui vise à retrouver les ruptures thématiques les plus importantes d'un texte. Étant donc supposés représenter chacun une thématique spécifique du texte, ces passages paraissent les plus à même de satisfaire les besoins d'un utilisateur ;
- *Window-based Passages* : séquences d'unités adjacentes du texte (généralement des mots), ces types de passages ont été proposés comme alternatives aux deux autres types de passages (face à l'inconsistance éventuelle des premiers et les difficultés d'obtention des deuxièmes). Ils peuvent être statiques, être définis une fois pour toutes lors de l'indexation, ou dynamiques, dépendant de la requête de l'utilisateur [4]. Dans la plupart des études, ces passages sont de taille fixe [Stanfill and Waltz, 1992; Zobel *et al.*, 1995]. Enfin, selon Callan et al., l'utilisation de passages se recouvrant les uns les autres peut être intéressant puisque cela permet de limiter la probabilité de diviser un bloc d'informations pertinentes en deux [Callan, 1994].

Dans [Kaszkiel and Zobel, 2001], des expériences ont été réalisées afin de définir quel type de passage permet le meilleur classement des documents en réponse à une requête utilisateur. Il s'est avéré que, contrairement aux attentes, les segments thématiques se sont révélés bien moins performants que des séquences de mots de taille fixe démarrant tous les 25 mots du texte [5] (séquences recouvrantes). Cette ob-

3. Cormack et al. [Cormack *et al.*, 1997] ont même montré que les jugements de pertinence que l'on pouvait établir en ne considérant qu'une unique séquence de 20 mots pour chaque document (la plus similaire à la requête) étaient très comparables à ceux établis dans *TREC* à partir des documents entiers. Ces observations mettent en évidence le fait que la pertinence des documents peut être efficacement estimée à partir d'un extrait restreint de leur texte.

4. Par exemple, Callan et al. font démarrer le premier passage d'un texte à la première occurrence d'un terme de la requête [Callan, 1994]

5. Dans [Kaszkiel and Zobel, 1997], des expériences ont montré qu'utiliser cet intervalle s'avérait aussi performant que si les séquences démarraient à chaque position (*i.e.,* chaque mot) du texte.

servation parait paradoxale puisqu'un découpage thématique des documents devrait permettre de délimiter de manière plus efficace la zone de texte la plus en rapport avec le sujet de l'utilisateur. En effet, si le nombre de séquences d'unités recouvrantes peut permettre de faire démarrer un passage au début du bloc contenant les informations les plus pertinentes du texte, leur longueur arbitrairement prédéfinie risque de les faire s'étendre sur une trop large zone de texte (ou au contraire de les faire se limiter à une partie trop restreinte de la zone de texte pertinente). Selon Kaszkiel et Zobel, cette observation paradoxale est principalement due aux grands écarts de taille existant entre les segments thématiques. Tel que l'on a pu le voir en section 1.3.2, les mesures de similarité existantes ne permettent pas de retourner tous les textes avec la même probabilité selon le nombre de termes qu'ils contiennent. Les séquences de mots, qui sont elles de taille fixe, sont favorisées puisque leurs estimations de pertinence se font alors de manière plus équitable. Après avoir essayé de corriger le problème en utilisant une mesure de normalisation des scores selon la taille des textes (la mesure *Pivoted Cosine*, formule 1.6), tentative qui n'a pas rencontré un réel succès, Kaszkiel et Zobel en ont alors conclu que les segments thématiques n'étaient pas bien adaptés aux approches de type *Passage Retrieval*. Pour nous, ce serait plutôt la mesure utilisée qui n'est pas bien adaptée aux segments thématiques : la mesure *Pivoted Cosine* a été conçue dans le but de permettre aux longs documents, qui ont, selon les expériences réalisées dans [Singhal *et al.*, 1996b], une plus grande probabilité de pertinence du fait du nombre de thématiques qu'ils abordent, d'être plus fréquemment retournés. Or, il n'y a pas de raison pour que les segments thématiques, qui sont supposés représenter chacun une unique thématique, soient sujets à ce phénomène. Il n'est pas du tout certain que les longs segments thématiques soient plus probablement pertinents que les segments plus courts. Par conséquent, pour s'assurer que les segments thématiques ne sont réellement pas adaptés aux approches de type *Passage Retrieval*, nous proposons de reproduire les expériences réalisées dans [Kaszkiel and Zobel, 2001] avec une mesure qui permette une vraie normalisation des scores selon la taille des textes, c'est à dire une mesure qui puisse donner à tous les textes, indépendamment de leur taille, la même espérance de score de similarité avec la requête. Il ne semble malheureusement pas exister de telle mesure... Ce chapitre commence donc par une analyse des différentes mesures de similarité pour tenter de mettre en place une fonction de normalisation des scores obtenus par la mesure *Cosine*.

6.2 Mesures de pertinence et longueur des textes

Bien que notre objectif soit la définition d'une mesure qui permette d'estimer la pertinence des segments thématiques de manière équitable, nous nous plaçons dans un premier temps dans un cadre de recherche d'information classique, où les analyses et comparaisons des mesures de similarité semblent plus aisées. Nous commençons cette étude par une comparaison de l'efficacité des mesures présentées en

section 1.3.2, c'est à dire les mesures *Cosine*, *Okapi*, *Inquery* et *Pivoted Cosine*. Le tableau 6.1 présente alors les résultats obtenus sur les quatre corpus présentés en section 3.2 avec les topics 1-50 de *TREC-1* pour lesquels on dispose de jugements de pertinence. Les mesures d'évaluation utilisées sont la *Mean Average Precision* (MAP), qui correspond à la moyenne des précisions moyennes Ap calculées pour chacune des requêtes (formule 3.3), et la precision au rang 100 ($P@100$), qui correspond à la moyenne des précisions calculées pour 100 documents retournés en réponse à chacune des 50 requêtes (formule 3.2) [6].

Cosine	Title		Narrative	
	MAP	P@100	MAP	P@100
ZIFF	0.176	0.123	**0.315**	0.209
AP	0.150	0.098	0.330	0.209
WSJ	0.151	0.128	0.383	0.275
FR	0.129	0.044	0.303	**0.119**

Okapi	Title		Narrative	
	MAP	P@100	MAP	P@100
ZIFF	**0.187**	0.140	0.309	**0.227**
AP	**0.175**	0.114	0.328	0.208
WSJ	0.215	0.158	0.405	**0.278**
FR	0.149	**0.062**	**0.312**	0.109

Inquery	Title		Narrative	
	MAP	P@100	MAP	P@100
ZIFF	0.137	0.115	0.111	0.127
AP	0.158	0.105	0.251	0.163
WSJ	0.187	0.153	0.260	0.193
FR	0.137	0.044	0.128	0.061

Pivoted Cosine	Title		Narrative	
	MAP	P@100	MAP	P@100
ZIFF	0.159	**0.166**	0.301	**0.227**
AP	0.174	**0.117**	**0.344**	**0.220**
WSJ	**0.223**	**0.165**	**0.408**	**0.278**
FR	**0.173**	0.060	0.307	0.090

TABLE 6.1 – Efficacité des mesures de pertinence

Tel qu'observé dans de nombreuses campagnes d'évaluation [Robertson *et al.*, 1992; Robertson *et al.*, 1993; Robertson *et al.*, 1995], la mesure *Okapi* semble obtenir des résultats significativement meilleurs que les mesures *Cosine* et *Inquery* sur la plupart des corpus. Selon [Singhal *et al.*, 1996b], ses meilleures performances sont majoritairement dues au fait que sa probabilité de sélection des documents à retourner est bien adaptée à la probabilité de pertinence des documents. Comme nous avons pu l'observer sur la figure 1.5, cette probabilité de pertinence semble croître avec la taille des documents. Par conséquent, Singhal et al. ont proposé d'adapter la mesure *Cosine* pour lui permettre d'adapter sa probabilité de sélection des documents en conséquence. Cette nouvelle mesure, la *Pivoted Cosine* [Singhal *et al.*, 1996b] que nous avons présentée plus en détails en section 1.3.2, semble être bien supérieure à toutes les autres mesures sur la plupart des corpus (excepté le corpus $ZIFF$).

Néanmoins, on est en droit de s'interroger sur les bienfaits réels d'une telle adaptation de la mesure *Cosine* : le fait de favoriser les longs documents ne comporte-t-il pas certains biais ? Ne risque-t-on pas d'écarter des petits documents qui auraient pu s'avérer très utiles pour les besoins de l'utilisateur ? Doit-on produire le plus long

6. Les valeurs en gras correspondent, pour chaque corpus, type de requête et critère d'évaluation, au meilleur score obtenu sur les quatre différentes mesures.

document possible pour avoir des chances d'être lu ?

Le fait que les documents les plus longs soient plus fréquemment pertinents, ce qui semble se vérifier dans nos expérimentations (figure 1.5), peut s'expliquer, tel que nous l'avons évoqué plus haut, par le plus grand nombre de thématiques qu'ils abordent. Traitant d'un plus grand nombre de sujets, ils sont susceptibles de répondre à un plus grand nombre requêtes. Cependant, cela doit-il impliquer de les retourner avec une plus grande probabilité ? Tout d'abord, notons que les documents longs sont généralement plus difficiles à interpréter, rechercher les informations pertinentes requiert un effort plus important dans ce type de document que dans des documents plus courts et donc, mieux ciblés. Selon Jacobs [Jacobs, 1992], les utilisateurs des systèmes préfèrent des réponses directes, concises et ciblées, aux questions qu'ils posent plutôt que de longs documents dans lesquels ils doivent trier les informations. Certes le volume d'informations intéressantes est peut-être plus important dans un long document, mais les difficultés que leur identification et leur intégration impliquent rendent les documents courts certainement préférables. Ici, une mesure qui adapte les probabilités de sélection des documents à leur probabilité de pertinence semble obtenir de meilleurs résultats par rapport aux jugements de pertinence réalisés lors de l'annotation des corpus. Cependant les jugements établis dans *TREC* sont binaires, ils ne donnent pas différents degrés de pertinence et donc aucune information n'est disponible sur l'utilisabilité des documents jugés pertinents. Si les appréciations réalisées comportaient de telles informations, il n'est pas sûr que les mesures telles que la *Pivoted Cosine* obtiennent d'aussi bons résultats. Par ailleurs, s'il est vrai que les mesures favorisant les longs documents sont susceptibles de retourner une plus grande proportion de documents pertinents, il n'en reste pas moins que les documents non pertinents retournés (puisqu'il y en a aussi) ont plus de chances d'être de longs documents, ce qui risque de compliquer considérablement la tâche de localisation des informations pertinentes.

D'un autre point de vue, la mise en place de techniques pour favoriser les longs documents pourrait se justifier par le fait que de tels documents ont plus de difficultés à être positionnés en tête de liste des résultats puisque, abordant un plus grand nombre de thématiques, les informations pertinentes qu'ils contiennent sont plus susceptibles d'être entourées de larges parties de texte fortement déconnectées des besoins de l'utilisateur. Tel que nous venons de l'exposer, les approches de *Passage Retrieval* permettent de redonner leurs chances à ces documents en ne considérant que la zone de texte la plus proche de la requête. Néanmoins, dans le cadre d'une recherche documentaire classique, nous pensons qu'il n'y a pas de raisons de favoriser les longs documents sur la seule base qu'ils peuvent contenir un hypothétique passage en relation avec la requête. Cela risque en effet de favoriser également les longs documents dans lesquels les termes de la requête sont dispersés sur l'ensemble du texte, ce qui n'est pas vraiment révélateur de l'existence d'informations pertinentes, alors que de petits documents, dans lesquels les termes de la requête sont susceptibles d'être plus proches les uns des autres, auraient pu s'avérer bien plus intéressants pour

les besoins de l'utilisateur.

Enfin, le fait que les longs documents soient plus fréquemment pertinents que les textes plus courts peut s'expliquer par un biais inhérent à la méthode de *pooling* utilisée par *TREC* (voir section 3.2) : puisque seuls les documents retournés par les systèmes participant à la campagne d'évaluation sont considérés par les annotateurs en charge des jugements de pertinence, il suffit que les systèmes soient légèrement enclins à retourner plutôt des documents longs pour qu'un plus grand nombre de documents pertinents soient identifiés. Zobel a en effet montré que de très nombreux documents pertinents n'ont pas été identifiés lors des campagnes d'évaluation [Zobel, 1998]. Selon nous, il ne serait pas surprenant que la majorité de ces documents soient relativement courts. Par ailleurs, le biais que nous mentionnons ici ne risque pas d'être résolu si de trop nombreux systèmes, à l'instar de la mesure *Pivoted Cosine*, tentent d'améliorer leurs performances, presque "artificiellement", en favorisant les longs documents sur la base d'observations statistiques faisant apparaître que de tels documents sont plus fréquemment pertinents. Il est en effet possible d'imaginer le scénario suivant :

1. Les systèmes participant à la première édition de *TREC* présentent pour la plupart le biais de favoriser quelque peu les longs documents ;
2. Utilisant la technique de *Pooling*, les annotateurs considèrent un nombre plus important de longs documents et donc identifient légèrement plus de longs documents pertinents que de courts ;
3. Considérant les jugements réalisés et observant que les longs documents des corpus semblent légèrement plus fréquemment pertinents que les courts, les systèmes participant à l'édition suivante adaptent leurs mesures de similarité pour retourner un plus grand nombre de longs documents et obtenir ainsi de meilleurs résultats ;
4. Utilisant la technique de *Pooling*, les annotateurs identifient bien plus de longs documents pertinents que de courts ;
5. Les participants de l'édition suivante adaptent encore un peu leurs mesures pour correspondre aux nouvelles données ;
6. Et ainsi de suite jusqu'à ne plus examiner que les documents les plus longs du corpus ...

Bien sûr, ce scénario est totalement fictif. Néanmoins, il serait intéressant d'étudier l'évolution du rapport longueur/pertinence au fil des éditions de *TREC*. Un moyen d'évaluer la distribution réelle des documents pertinents selon les tailles des textes pourrait consister à appliquer la méthode d'extrapolation utilisée par Zobel [Zobel, 1998] (pour estimer le nombre total de documents pertinents dans le corpus, voir section 3.2) à différentes tranches de taille de documents, permettant ainsi d'estimer la fréquence de pertinence pour chacune d'entre elles. De cette manière, on pourrait avoir une meilleure idée de la mesure dans laquelle les documents longs sont réellement plus fréquemment pertinents que les courts.

D'une manière plus générale, nous pensons que le fait de favoriser la sélection des longs documents n'est pas le meilleur moyen d'aider l'utilisateur dans sa tâche de recherche d'informations. Afin de s'en rendre compte de manière quantitative, nous proposons de substituer au critère de précision classique, qui considère la proportion de documents pertinents dans l'ensemble des documents retournés, un critère de précision des termes retournés, qui considère le ratio de termes appartenant aux documents pertinents dans l'ensemble des termes appartenant aux documents retournés. Aini, avec $Pert(D_i)$ une fonction $Pert : \mathcal{D} \rightarrow \{0, 1\}$ retournant 1 si le document D_i est pertinent (0 sinon) et $|D_i|$ le nombre de termes contenus par le documents D_i, la précision des termes $TP@r$ au rang r s'obtient par :

$$TP@r = \frac{\sum_{i=1}^{r} Pert(D_i) \times |D_i|}{\sum_{i=1}^{r} |D_i|} \tag{6.1}$$

L'utilisation d'un tel critère sur un système donné permet de rendre compte d'un ratio gain/coût, c'est à dire du volume d'informations pertinentes collectées pour le volume d'efforts à fournir pour les localiser. Nous reproduisons alors les mêmes expérimentations que celles qui ont conduit aux résultats présentés dans le tableau 6.1 en appliquant notre nouvelle mesure aux critères MAP (qui devient $MATP$) et $P@100$ (qui devient alors $TP@100$). Les résultats obtenus sont présentés dans le tableau 6.2.

Cosine	Title		Narrative	
	MATP	TP@100	MATP	TP@100
ZIFF	0.210	0.149	**0.379**	0.264
AP	0.176	0.121	**0.355**	**0.227**
WSJ	0.203	**0.190**	**0.429**	**0.321**
FR	0.175	**0.087**	**0.384**	**0.168**

Okapi	Title		Narrative	
	MATP	TP@100	MATP	TP@100
ZIFF	**0.225**	**0.179**	0.347	**0.292**
AP	**0.182**	**0.126**	0.335	0.214
WSJ	**0.239**	**0.190**	0.417	0.286
FR	0.174	0.080	0.331	0.125

Inquery	Title		Narrative	
	MATP	TP@100	MATP	TP@100
ZIFF	0.124	0.117	0.081	0.104
AP	0.158	0.105	0.237	0.143
WSJ	0.184	0.147	0.221	0.142
FR	**0.181**	0.070	0.068	0.034

Pivoted Cosine	Title		Narrative	
	MATP	TP@100	MATP	TP@100
ZIFF	0.138	0.153	0.294	0.204
AP	0.172	0.116	0.346	0.216
WSJ	0.220	0.157	0.406	0.268
FR	0.116	0.042	0.299	0.074

TABLE 6.2 – Informations collectées *Vs.* Longueur des textes

Par l'utilisation de ce nouveau critère, qui prend en compte la longueur des textes, la mesure *Pivoted Cosine* paraît bien moins bénéfique que lors de son évaluation par les critères classiques. En fait, elle ne surpasse, pour le critère $MATP$, les résultats de la mesure *Cosine* que dans un seul cas, lors de recherches dans le corpus WSJ avec de petites requêtes. Dans tous les autres cas, le plus fort taux de documents

retournés ne suffit pas à équilibrer les pénalités induites par la longueur des documents non pertinents retournés. Et encore, le critère utilisé ne prend pas en compte le fait que seule une partie restreinte des documents pertinents peut en fait être réellement intéressante. Si cela avait été le cas, les résultats auraient certainement été encore bien pires puisque l'on peut supposer que le nombre de termes appartenant à des passages déconnectés du besoin de l'utilisateur est plus élevé dans les documents pertinents retournés par cette mesure que dans ceux retournés par la mesure *Cosine*. Nous pensons alors que les bons résultats obtenus par cette mesure ne reflétaient pas ses réelles performances. Par ailleurs, si l'on peut encore considérer que cette mesure est applicable à la recherche documentaire, puisqu'elle permet tout de même d'obtenir un plus fort taux de documents pertinents, elle ne l'est certainement pas dans les approches de type *Passage Retrieval*, et plus particulièrement pour la recherche de segments thématiques, qui, par leur définition, sont supposés représenter chacun une thématique unique, et donc n'ont pas de raison d'être plus probablement pertinents lorsque le texte qu'ils contiennent est long. Nous procédons alors maintenant à une analyse de l'influence que peut avoir la longueur des textes sur les scores de similarité déterminés par les mesures *Cosine*, *Okapi* et *Inquery*, afin de trouver le moyen d'estimer la pertinence des textes de manière plus équitable.

6.2.1 Impacts de la longueur des textes sur les estimations de pertinence

L'étude proposée ici ressemble quelque peu à celle réalisée dans [Singhal *et al.*, 1996b], puisque nous cherchons également à observer les tendances des mesures de similarité face aux variations de longueur des textes. Néanmoins, plutôt que d'étudier l'influence de la longueur d'un texte sur sa probabilité de sélection (voir section 1.3.2), nous proposons d'observer ici l'impact de ces variations de longueur sur l'espérance des scores de similarité attribués par les différentes mesures. À la manière de [Singhal *et al.*, 1996b], nous formons alors des groupes de documents en fonction de leur taille (en nombre de termes significatifs distincts) et nous capturons les tendances de chaque groupe lors de recherches de documents correspondant à différentes requêtes. Pour chacune des requêtes utilisées et chacune des trois mesures étudiées, nous enregistrons le score moyen obtenu par les documents de chaque groupe [7]. Nous considérons alors que, pour une mesure de similarité donnée, l'espérance du score que peut obtenir un document correspond à la moyenne des scores moyens obtenus par le groupe auquel il appartient sur l'ensemble des requêtes utilisées.

Par ailleurs, Chung et al. [Chung *et al.*, 2006] ayant fait remarquer que la longueur des requêtes pouvaient également avoir une influence significative sur les mesures,

7. En ne considérant cependant que les scores des documents ayant au moins un terme en commun avec la requête ($Sim(D, R) > 0$). Nous nous intéressons ici au positionnement des documents dans la liste de résultats. Or, ces documents n'ont aucune chance d'y figurer, ils ne sont donc pas concernés par l'étude. Par ailleurs, le nombre de documents ne possédant aucun terme en commun avec la requête étant bien plus important dans les groupes de petits documents que dans les grands, leur prise en considération aurait certainement biaisé l'étude, diminuant considérablement la moyenne des scores des petits documents.

nous intégrons les tailles des requêtes à notre étude. Les jeux de requêtes de *TREC* ne présentant pas un nombre de représentants de différentes tailles suffisamment important, nous choisissons de travailler avec des requêtes que nous construisons "artificiellement" : pour chaque taille de requête entre 1 et 200 termes significatifs uniques (variant par pas de 5 termes), 1000 requêtes sont alors automatiquement produites en choisissant, pour chacune d'entre elles, chaque terme dans un document de plus de 300 termes sélectionné aléatoirement (parmi les documents des quatre corpus présentés en section 3.2) [8]. Les espérances de scores attribués par les mesures de similarité sont alors calculées indépendamment pour chacune de ces 41 tailles de 1000 requêtes.

Les graphes de la figure 6.1 donnent l'évolution, selon les tailles des documents (ou plutôt les tailles moyennes des groupes de documents), de l'espérance des scores attribués par les mesures de similarité [9] (moyennes des scores obtenus) observée sur le corpus *ZIFF* (section 3.2) pour des requêtes de 1, 10, 50 et 200 termes uniques. On peut tout d'abord noter qu'aucune des mesures ne semble attribuer la même

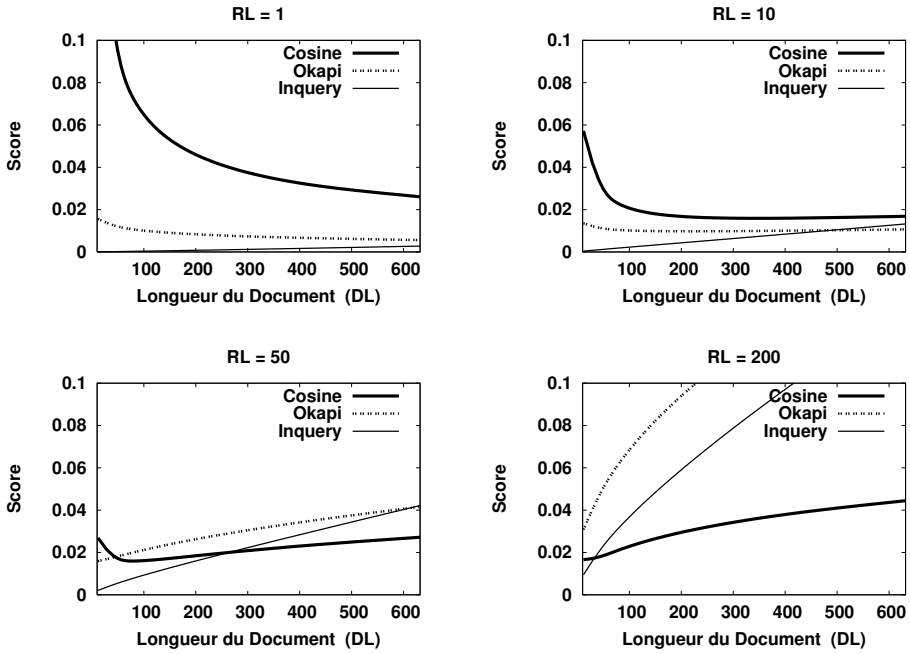

FIGURE 6.1 – Tendances des scores de pertinence

espérance de score selon les tailles des documents et des requêtes. La taille de la

8. Le fait de ne choisir que des termes appartenant à un même document permet d'obtenir des requêtes plus probablement cohérentes que si les termes utilisés provenaient de différentes sources.

9. Les scores des mesures *Okapi* et *Inquery* ont été divisés par 100 afin de pouvoir les faire figurer dans les mêmes graphes que ceux de la mesure *Cosine*.

requête utilisée semble effectivement avoir un impact considérable sur les scores de similarité des documents : alors que les mesures *Cosine* et *Okapi* tendent à favoriser les petits documents lors de recherches utilisant des requêtes courtes, ces mêmes mesures semblent préférer les longs documents lorsque la taille des requêtes augmente [10]. Cette inversion de préférence peut être expliquée par le fait que l'avantage qu'ont les petits documents, c'est à dire une plus faible probabilité de contenir des termes n'appartenant pas à la requête (lorsqu'au moins un des termes de la requête est présent dans leur texte), est surpassé, lorsque la taille de la requête augmente, par celui que présentent les plus longs documents, c'est à dire une plus forte probabilité de contenir tous les termes de la requête. En supposant que la probabilité de sélection soit corrélée avec l'espérance mathématique des scores de similarité attribués, l'observation réalisée dans [Chung *et al.*, 2006], que l'angle d'inclinaison utilisé par la mesure *Pivoted Cosine* doit dépendre de la taille de la requête considérée, paraît alors confirmée par les expérimentations conduites ici. Cependant, au regard de la forme des courbes, il semble clair qu'une normalisation efficace des scores ne peut pas être obtenue en appliquant une fonction de transformation linéaire des mesures telle que celle proposée dans la mesure *Pivoted Cosine*. De plus, pour les très longues requêtes, la probabilité de sélection des longs documents dépasse largement leur probabilité de pertinence. Contrairement à ce qui est fait dans [Chung *et al.*, 2006], leur score de similarité avec la requête devrait alors être diminué.

Afin d'étudier l'évolution des écarts des scores de similarité à la moyenne selon les tailles des documents, nous avons considéré, pour chaque requête, le score maximal obtenu dans chaque groupe. La moyenne de ces scores maximaux sur l'ensemble des requêtes de chaque taille a fait apparaître que l'espérance du score du meilleur document de chaque groupe était soumise aux mêmes variations que l'espérance de scores pour l'ensemble des documents des groupes. Les écarts des scores à la moyenne semblent alors relativement constants. Si nous disposions d'une fonction de modélisation de l'espérance mathématique $E(D_L, R_L)$ des scores attribués par une mesure aux documents selon le nombre de termes distincts D_L qu'ils contiennent et le nombre de termes distincts R_L que comporte la requête considérée, nous pourrions proposer la normalisation des scores suivante, qui vise à donner à tous les scores de similarité entre documents et requêtes une même espérance de 0.5 :

$$NormSim(D, R) = \frac{1 + Sim(D, R) - E(D_L, R_L)}{2} \tag{6.2}$$

Une telle fonction de normalisation permettrait alors de réaliser des estimations de pertinence bien plus équitables, puisqu'attribuant la même probabilité de sélection à tous les documents, quelle qu'en soit la taille. La section suivante s'attache à la recherche d'une telle fonction de modélisation de l'espérance des scores de similarité entre documents et requêtes attribués par la mesure *Cosine*.

10. Avec la mesure *Inquery*, cette inversion n'est pas observée, la préférence pour les longs documents est déjà bien marquée quand la requête ne contient que peu de termes.

6.2.2 Normalisation des scores de pertinence par régression statistique

Nous proposons de réaliser la modélisation de l'espérance des scores attribués par la mesure *Cosine* par application de techniques de régression statistique [Yule, 1897] [11] sur les distributions de similarités observées sur le corpus $ZIFF$ (figure 6.1). Après avoir déterminé les variables et coefficients de notre modèle, nous en expérimentons la validité sur différents corpus. Enfin, nous évaluons son impact sur les performances des systèmes de recherche d'information.

Une équation de régression statistique vise à exprimer les relations existant entre différentes variables observées :

$$y = a_0 + a_1 \times x_1 + a_2 \times x_2 + \cdots + a_p \times x_p \tag{6.3}$$

où y est la variable pour laquelle on construit le modèle, $x_1 \ldots x_p$ correspondent aux variables "prédictives" du problème et $a_0 \ldots a_p$ représentent les paramètres (ou coefficients) du modèle.

La première étape de l'établissement d'un modèle consiste à déterminer les variables qu'il doit comporter, c'est à dire rechercher les variables qui permettent d'expliquer au mieux les différentes valeurs observées pour la sortie y du modèle. Dans notre cas, la sortie, c'est à dire la variable que l'on cherche à estimer, correspond à l'espérance du score attribué par la mesure *Cosine*. Les variables prédictives, quant à elles, doivent dépendre des longueurs du document et de la requête concernés par le calcul de similarité. Notons que nos deux longueurs de textes (requête et document) ne peuvent pas être considérées comme deux variables indépendantes car, tel qu'observé sur la figure 6.1, la longueur du document considéré n'a pas le même impact selon la taille de la requête à laquelle il est comparé. Face à la difficulté de trouver directement une modélisation de la fonction d'espérance selon les différentes tailles de documents et de requêtes, nous avons décidé dans un premier temps de diviser le problème en autant de sous-problèmes que de tailles de requête. C'est à dire que nous cherchons à définir un ensemble d'équations $E_{R_L}(D_l)$ correspondant chacune à la modélisation, pour une taille de requête particulière, de l'espérance du score de similarité attribué par la mesure *Cosine* selon différentes tailles de document.

La recherche des meilleures variables est réalisée en considérant un coefficient de corrélation $R^2_{y,x_1x_2\ldots x_p}$, qui traduit l'intensité des relations existant entre les p variables prédictives expérimentées et la sortie du problème y selon un ensemble de n observations [Yule, 1897]. Avec un modèle comportant deux variables prédictives,

11. La régression statistique a déjà été expérimentée en recherche d'information par le passé (par exemple dans [Cooper *et al.*, 1992]), mais le point de vue est ici différent : alors que nous cherchons à normaliser une mesure existante selon les différentes tailles des textes comparés, l'objectif des approches proposées au préalable était de déterminer une nouvelle mesure de similarité basée sur un apprentissage des probabilités de pertinence selon les tailles des documents considérés, ce qui correspond aux mesures, critiquées dans les sections précédentes, qui cherchent à favoriser les documents dont le nombre de termes semble correspondre à la taille pour laquelle il y a le plus de représentants pertinents dans le corpus.

ce coefficient correspond à :

$$R^2_{y,x_1x_2} = \frac{R^2_{y,x_1} + R^2_{y,x_2} - 2R_{y,x_1}R_{y,x_2}R_{x_1,x_2}}{1 - R^2_{x_1,x_2}} \tag{6.4}$$

où $R_{a,b}$ correspond à un coefficient de corrélation entre deux variables a et b, c'est à dire la co-variance des variables a et b divisée par les variances individuelles de a et de b. Plus le score obtenu par ce coefficient est élevé, plus le modèle produit a des chances de réaliser de bonnes approximations de y.

De nombreux ensembles de variables ont été expérimentés sur notre corpus $ZIFF$ d'entraînement, et nous avons sélectionné le modèle qui permet l'obtention du meilleur coefficient de corrélation :

$$E_{R_L}(D_L) = a_0 + a_1 \times \ln(D_L) + a_2 \times \ln(\ln(D_L) + 1) \tag{6.5}$$

où D_L et R_L correspondent au nombre de termes uniques dans le document et la requête respectivement.

Les variables prédictives étant définies, le problème consiste maintenant à déterminer les coefficients a_0, a_1 et a_2 à leur associer. Pour ce faire, nous utilisons la méthode des moindres carrés [Björck, 1996], qui fait l'hypothèse que la courbe exprimant au mieux la variable à estimer est celle dont la somme des carrés des écarts aux valeurs observées est la plus faible. Dans notre problème, cela revient à résoudre le système d'équations suivant :

$$\begin{cases} a_0 \times n + a_1 \times \sum_{i=1}^{n} x_{1,i} + a_2 \times \sum_{i=1}^{n} x_{2,i} = \sum_{i=1}^{n} y_i \\ a_0 \times \sum_{i=1}^{n} x_{1,i} + a_1 \times \sum_{i=1}^{n} x_{1,i}^2 + a_2 \times \sum_{i=1}^{n} x_{1,i}x_{2,i} = \sum_{i=1}^{n} x_{1,i}y_i \\ a_0 \times \sum_{i=1}^{n} x_{2,i} + a_1 \times \sum_{i=1}^{n} x_{1,i}x_{2,i} + a_2 \times \sum_{i=1}^{n} x_{2,i}^2 = \sum_{i=0}^{n} x_{2,i}y_i \end{cases} \tag{6.6}$$

où n correspond au nombre d'observations utilisées pour l'entraînement du modèle, y_i à la i-ième valeur de y observée et $x_{1,i}$ et $x_{2,i}$ aux valeurs des deux variables prédictives lors de la considération de y_i.

La figure 6.2 présente les valeurs des paramètres obtenues pour les différentes fonctions $E_{R_L}(D_L)$ (chaque point correspondant alors à une taille de requête donnée). Une modélisation de l'évolution de ces trois coefficients a été réalisée par application de la méthodes des moindres carrés pour établir la fonction finale $E(D_L, R_L)$ qui combine les multiples fonctions $E_{R_L}(D_L)$. En utilisant les 41 équations $E_{R_L}(D_L)$ dont on dispose (41 équations pour 41 tailles de requête étudiées entre 1 et 201 termes uniques), on obtient la fonction d'estimation de l'espérance

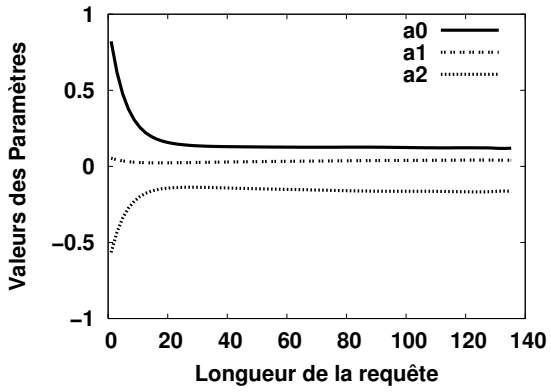

FIGURE 6.2 – Valeurs des coefficients

des scores attribués par la mesure *Cosine* suivante :

$$\begin{aligned} E(D_L, R_L) = \quad & (1.00586 + 0.18685 \times \ln(R_L) - 1.02757 \times \ln(\ln(R_L) + 1)) \\ & + \ln(D_L) \times \\ & (0.09036 + 0.02671 \times \ln(R_L) - 0.10388 \times \ln(\ln(R_L) + 1)) \\ & + \ln(\ln(D_L) + 1) \times \\ & (-0.77143 - 0.16194 \times \ln(R_L) + 0.803 \times \ln(\ln(R_L) + 1)) \end{aligned} \tag{6.7}$$

Alors que le coefficient R^2_{y,x_1x_2} évalue les relations existant entre les différentes variables, le coefficient $R^2_{y,\hat{y}}$, quant à lui, considère le degré de corrélation entre valeurs observées y et valeurs estimées $\hat{y}$:

$$R^2_{y,\hat{y}} = 1 - \frac{\text{Variabilité Non Expliquée}}{\text{Variabilité Totale}} = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \tag{6.8}$$

où N correspond au nombre total d'observations disponibles et $\bar{y}$ à la moyenne des valeurs observées sur cet ensemble pour la variable y à estimer. Ce coefficient nous permet d'évaluer l'efficacité du modèle sur le corpus d'entrainement, ainsi que sa validité sur d'autres corpus.

Afin de déterminer la taille de l'ensemble d'observations requises pour entraîner efficacement le modèle établi, ainsi que d'évaluer sa capacité à être appliqué à d'autres corpus que celui sur lesquels ses paramètres ont été optimisés, différents sous-ensembles de l'ensemble des N observations disponibles ont été expérimentés. Dans ce qui suit nous notons alors $T(c, x)$ un ensemble d'entrainement constitué des observations obtenues avec les x premières tailles de requêtes sur le corpus c.

Une coefficient supérieur à 0.8 est généralement considéré comme traduisant une forte corrélation et dénote donc un modèle bien adapté au problème concerné.

$R^2_{y,\hat{y}}$	$E(D_L, R_L)$			
	ZIFF	AP	WSJ	FR
$T(ZIFF, 5)$	0.938	0.931	0.909	0.719
$T(ZIFF, 10)$	0.981	0.981	0.971	0.89
$T(ZIFF, 20)$	0.992	0.984	0.984	0.937
$T(ZIFF, 41)$	0.994	0.976	0.99	0.959
$T(AP, 5)$	0.923	0.915	0.892	0.661
$T(AP, 10)$	0.954	0.95	0.93	0.756
$T(AP, 20)$	0.974	0.98	0.958	0.845
$T(AP, 41)$	0.987	0.993	0.977	0.913

$R^2_{y,\hat{y}}$	$E(D_L, R_L)$			
	ZIFF	AP	WSJ	FR
$T(WSJ, 5)$	0.948	0.949	0.931	0.859
$T(WSJ, 10)$	0.976	0.969	0.975	0.909
$T(WSJ, 20)$	0.987	0.966	0.992	0.946
$T(WSJ, 41)$	0.988	0.948	0.995	0.968
$T(FR, 5)$	0.948	0.931	0.953	0.909
$T(FR, 10)$	0.971	0.929	0.967	0.944
$T(FR, 20)$	0.977	0.897	0.951	0.982
$T(FR, 41)$	0.972	0.872	0.934	0.972

TABLE 6.3 – Coefficients de corrélation

Les résultats présentés dans la table 6.3 montrent alors que le modèle établi ici permet une bonne approximation de l'espérance de similarité entre documents et requêtes selon la taille des textes comparés. Il semble relativement robuste puisque des paramètres optimisés sur un ensemble complet de N observations réalisées sur un corpus donné (quel qu'il soit), permet l'obtention de coefficients de corrélation quasiment aussi élevés sur les autres corpus. Par ailleurs, il semble que l'ensemble d'observations utilisé pour optimiser les coefficients du modèle n'a pas besoin d'être très grand puisque, selon le tableau 6.3, les paramètres entrainés sur les 20 premières tailles de requêtes paraissent être quasiment d'aussi bons approximateurs que ceux ayant utilisé l'ensemble des observations dans sa totalité. Selon les résultats, le modèle permettant les meilleures estimations semble être celui dont les paramètres ont été optimisés sur l'ensemble d'observations réalisées sur le corpus $ZIFF$, c'est à dire le modèle correspondant à l'equation 6.7.

Le tableau 6.4 présente les résultats d'un système de recherche d'information utilisant notre mesure, notée *NCosine* (pour *Normalized Cosine*), qui réalise une normalisation des scores de la mesure *Cosine* par application des équations 5.19 et 6.7 [12]. En comparant les données reportées dans ce tableau avec celles des ta-

NCosine	Title				Narrative			
	MAP	P@100	MATP	TP@100	MAP	P@100	MATP	TP@100
ZIFF	**0.192**	0.146	**0.225**	**0.180**	0.306	0.207	0.371	0.271
AP	0.174	**0.118**	**0.189**	**0.132**	0.333	0.216	**0.361**	**0.239**
WSJ	0.202	0.159	0.235	0.183	**0.367**	0.272	**0.429**	**0.337**
FR	0.154	0.061	**0.198**	**0.093**	0.234	0.106	0.335	0.177

TABLE 6.4 – Efficacité de la mesure *Cosine* normalisée par régression statistique

bleaux 6.1 et 6.2, on s'aperçoit que, lors de l'utilisation de requêtes relativement

12. Les valeurs en gras correspondent aux scores, obtenus par la mesure *NCosine*, étant supérieurs à ceux obtenus par l'ensemble des quatre autres mesures de pertinence (voir tables 6.1 et 6.2).

courtes (*Title*), notre mesure *NCosine* permet l'obtention de résultats significativement meilleurs [13] que ceux de la mesure *Cosine*, quel que soit le critère ou le corpus considéré. Avec des requêtes plus longues, la mesure *Cosine* tend à favoriser les longs documents (voir figure 6.1) et, étant donné que ces documents sont plus probablement pertinents, cette mesure obtient de très bons résultats. Néanmoins, notre mesure permet la sélection de documents courts (ce qui n'est généralement pas le cas avec les quatre autres mesures). Les résultats qu'elle obtient alors avec le critère de précision de termes retournés (que nous avons présenté en section 6.2) montrent que les documents qu'elle retourne sont plus facilement utilisables que ceux retournés par les autres mesures. Tel que discuté précédemment, nous pensons que les mesures favorisant la sélection des longs documents obtiennent de bons résultats pour de mauvaises raisons. Le critère de précision des termes retournés limite le biais permettant à ces mesures d'obtenir des bons résultats. Par ailleurs, nous pensons que notre mesure, obtenant déjà les meilleurs résultats sur les corpus étudiés, montrerait une dominance plus nette sur des corpus observant une distribution plus uniforme des jugements de pertinence selon les tailles de documents.

6.3 Types de segments et performances des systèmes

Tel qu'énoncé en section 6.1, nous proposons de reproduire les expériences réalisées dans [Kaszkiel and Zobel, 2001] afin de déterminer de manière définitive si les segments thématiques ne sont réellement pas adaptés aux approches de type *Passage Retrieval*. Dans la section précédente, nous avons défini une mesure de similarité qui cherche à estimer la pertinence des textes de manière équitable, quelle que soit la taille du texte et de la requête comparés. Certes, cette mesure n'a pas été mise en place dans un cadre de *Passage Retrieval* mais il est possible de considérer que les segments thématiques ne sont jamais que des petits textes ayant plus ou moins les mêmes propriétés que des documents ciblés sur un sujet relativement précis. Il n'y a alors pas de raison que les observations utilisées pour la constitution du modèle sur lequel se base notre mesure *NCosine* ne soient pas valides dans le cadre des approches qui nous intéressent ici.

Les principales tentatives d'explication des mauvaises performances des approches de type *Passage Retrieval* utilisant des segments thématiques reposent sur la grande diversité de tailles que présente l'ensemble des segments considérés. Nous pensons que notre mesure *NCosine*, tentant de passer outre les biais induits par les différences entre tailles des textes comparés, peut permettre à ces approches d'obtenir de bien meilleurs résultats. Dans un premier temps, nous présentons les différents types de passages auxquels nous comparons les segments thématiques dans un contexte de *Passage Retrieval*. Enfin, nous présentons et tentons d'analyser les résultats obtenus par les différentes approches.

13. Cette affirmation a été vérifiée par un test de Student.

6.3.1 Types de segments étudiés

Tel qu'énoncé en première partie de ce chapitre, les principaux passages auxquels Kaszkiel et Zobel comparent les segments thématiques dans un contexte de *Passage Retrieval* sont des séquences de mots de taille fixe. Nous considérons alors deux types de séquences de mots dans nos expérimentations : dans une approche que nous notons M_N, les passages considérés sont des séquences de N mots démarrant tous les N mots consécutifs du texte (passages non recouvrants), et dans une autre approche que nous notons RM_N, les passages considérés sont des séquences de N mots démarrant tous les 25 mots consécutifs (passages recouvrants). Afin de prendre en compte les mots situés en fin de textes, nous ajoutons aux éléments considérés par ces deux approches les passages correspondant aux N derniers mots de chaque document.

Selon Kaszkiel et Zobel [Kaszkiel and Zobel, 2001], le principal avantage de ces deux types de passages réside dans le fait que les ensembles de textes manipulés sont de tailles homogènes, ce qui écarte les difficultés liées aux comparaisons de scores de pertinence entre textes de tailles différentes. Néanmoins, étant donné le schéma de pondération en $tf * idf$ utilisé, un grand nombre de mots, puisque très génériques et donc présents dans un très grand nombre de documents (ou passages), risquent de ne peser que très peu dans les calculs de similarité. Le score de pertinence final est alors quasiment le même que l'on considère ces mots ou non (en les supprimant à l'aide d'une *stop-list*, voir section 1.2.2). Par conséquent, dans le cas de passages définis comme des séquences de mots, il est très probable qu'un grand nombre de mots ne pèsent pas dans l'estimation de la pertinence de chaque passage. Les problèmes liés aux difficultés de comparer des scores de pertinence obtenus par des textes de tailles différentes subsistent alors quelque peu dans les approches manipulant de tels passages. Nous allons donc plus loin dans la détermination de passages de tailles homogènes en introduisant deux nouveaux types d'approches se basant sur des passages définis comme des séquences de termes significatifs (termes utilisés pour l'indexation des textes) : alors que dans T_N, des passages de N termes consécutifs sont déterminés tous les N termes du texte, dans RT_N, les passages correspondent à des séquences de N termes consécutifs pris tous les 25 mots du texte (l'intervalle utilisé correspond encore ici à un nombre de mots afin d'obtenir un nombre de passages comparable à celui considéré par l'approche RM_N).

Lors des expérimentations présentées dans la section 6.2.2, nous nous sommes aperçus que l'espérance des scores de similarité avec la requête était plus fortement corrélée avec le nombre de termes distincts des textes qu'avec leur nombre total de termes significatifs. Pour en finir avec la définition de passages permettant d'écarter les biais relatifs à la taille des textes manipulés, nous introduisons deux derniers types d'approches qui considèrent des séquences de termes distincts (les séquences contiennent alors toutes N termes différents) : U_N considère des séquences de N termes distincts consécutifs déterminées les unes après les autres et RU_N utilise des séquences de N termes distincts définis tous les 25 mots du texte.

Corpus	ZIFF	AP	WSJ	FR
Documents	6939	6740	7898	4151
Mots par document	445	479	557	2879
Termes par document	213	236	269	1495
Termes uniques par document	119	153	165	350
Mots des 1000 documents les + courts	89	154	67	238
Mots des 1000 documents les + longs	1822	894	1513	9521
Termes des 1000 documents les + courts	46	80	36	137
Termes des 1000 documents les + longs	840	434	717	4834
Termes uniques 1000 documents les + courts	36	59	28	88
Termes uniques des 1000 documents les + longs	387	275	411	838
Segments thématiques	12977	12806	17288	21997
Requêtes avec au moins un document pertinent	39	43	47	27
Documents pertinents par requête	25.89	20.22	29.21	12.22

TABLE 6.5 – Statistiques des corpus réduits

6.3.2 Comparaison des approches

Bien qu'une évaluation des approches de type *Passage Retrieval* considérant directement les passages plutôt que les documents les contenant aurait pu paraître intéressante, nous adoptons ici la même méthodologie d'évaluation que celle utilisée dans [Kaszkiel and Zobel, 2001] [14], c'est à dire que l'évaluation se fait sur une liste de documents ordonnée selon la similarité à la requête du meilleur passage de chacun des documents. Étant donné le nombre de passages induits par les approches utilisant des séquences d'unités recouvrantes, l'utilisation des corpus présentés en section 3.2 aurait été difficile. Nous avons alors choisi de restreindre le nombre de documents contenus par chacun des corpus en ne conservant que les 200 premiers documents retournés en réponse à chacun des 50 topics 1-50 de *TREC* par un système de recherche d'information classique (utilisant la mesure *Cosine*). Les statistiques de ce nouveau corpus sont présentées dans le tableau 6.5 [15]. Dans les résultats expérimentaux présentés par la suite, seules les requêtes possédant au moins un document pertinent dans le corpus sont considérées.

Le tableau 6.6 présente les résultats des différentes approches obtenus selon le critère habituel de *Mean average precision* (MAP) sur les quatre corpus réduits. Trois mesures de similarité sont expérimentées ici : la mesure *Cosine* (Cos), la mesure *Pivoted Cosine* [16] (Piv) et notre mesure de *Cosine* normalisé ($NCos$). Toutes utilisent un facteur *idf* qui dépend, pour chaque terme, du nombre de passages le contenant. Pour chaque approche manipulant des séquences d'unités, de nombreuses tailles

14. De plus, du fait de la multitude de types de passages à envisager, le nombre de jugements requis et la difficulté parfois rencontrée pour définir la pertinence d'un passage sorti de son contexte, il n'existe pas (ou trop peu) de ressources pouvant permettre d'envisager l'évaluation directe des passages retournés par un système.

15. Si les nombres de documents présents dans chaque corpus ne sont pas égaux à 10000 (50 requêtes × 200 documents), c'est que certains documents ont été retournés en réponse à plusieurs requêtes et que nous n'en n'avons conservé qu'une seule occurrence.

16. Où le pivot correspond à la moyenne des facteurs de normalisation calculés sur l'ensemble des passages du corpus.

FR							
Title	*Cos*	*Piv*	*NCos*	Narrative	*Cos*	*Piv*	*NCos*
Doc	0.131	0.229	0.159	*Doc*	0.327	0.329	0.318
M_{500}	0.230	0.222	0.248	M_{400}	0.332	0.327	0.346
T_{400}	0.232	0.223	0.247	T_{350}	0.341	0.332	0.346
U_{350}	0.234	0.223	0.250	U_{200}	0.344	0.332	0.348
RM_{450}	0.243	0.226	0.268	RM_{350}	0.343	0.339	0.357
RT_{350}	0.262	0.238	0.269	RT_{200}	0.359	0.347	0.359
RU_{250}	0.264	0.239	0.272	RU_{150}	0.361	0.342	0.360
Them	0.187	0.207	0.270	*Them*	0.322	0.326	0.354
WSJ							
Title	*Cos*	*Piv*	*NCos*	Narrative	*Cos*	*Piv*	*NCos*
Doc	0.206	0.279	0.245	*Doc*	0.416	0.385	0.400
M_{300}	0.228	0.243	0.252	M_{250}	0.409	0.390	0.411
T_{250}	0.235	0.246	0.255	T_{150}	0.411	0.390	0.411
U_{100}	0.240	0.247	0.255	U_{100}	0.413	0.393	0.412
RM_{300}	0.249	0.265	0.274	RM_{200}	0.417	0.403	0.416
RT_{200}	0.260	0.267	0.276	RT_{100}	0.419	0.402	0.418
RU_{100}	0.262	0.265	0.278	RU_{100}	0.420	0.405	0.418
Them	0.219	0.263	0.273	*Them*	0.410	0.390	0.416

ZIFF							
Title	*Cos*	*Piv*	*NCos*	Narrative	*Cos*	*Piv*	*NCos*
Doc	0.205	0.218	0.236	*Doc*	0.354	0.253	0.355
M_{400}	0.223	0.210	0.236	M_{300}	0.358	0.298	0.363
T_{250}	0.226	0.208	0.237	T_{200}	0.361	0.301	0.365
U_{150}	0.227	0.209	0.237	U_{100}	0.361	0.300	0.364
RM_{400}	0.230	0.216	0.242	RM_{300}	0.377	0.310	0.379
RT_{200}	0.243	0.214	0.244	RT_{150}	0.379	0.310	0.380
RU_{150}	0.244	0.214	0.244	RU_{100}	0.380	0.306	0.379
Them	0.213	0.214	0.243	*Them*	0.340	0.294	0.377
AP							
Title	*Cos*	*Piv*	*NCos*	Narrative	*Cos*	*Piv*	*NCos*
Doc	0.172	0.198	0.191	*Doc*	0.326	0.318	0.330
M_{300}	0.185	0.186	0.186	M_{300}	0.327	0.318	0.328
T_{200}	0.184	0.184	0.190	T_{150}	0.329	0.321	0.331
U_{100}	0.186	0.187	0.190	U_{150}	0.330	0.321	0.331
RM_{300}	0.193	0.194	0.197	RM_{250}	0.334	0.329	0.336
RT_{200}	0.193	0.193	0.196	RT_{150}	0.335	0.330	0.336
RU_{100}	0.194	0.194	0.197	RU_{100}	0.338	0.332	0.338
Them	0.172	0.187	0.198	*Them*	0.307	0.302	0.335

TABLE 6.6 – Résultats des approches selon les trois mesures de pertinence

de séquences ont été expérimentées (toutes les tailles entre 50 et 600, par paliers de 50). Le tableau 6.6 ne présente que les séquences ayant obtenu les meilleurs résultats. Enfin, l'approche *Doc* correspond à une recherche d'information classique (manipulant des documents) et *Them* correspond à l'approche utilisant des segments thématiques [17] (les autres notations, qui concernent les approches par séquences d'unités, sont définies dans la section précédente).

Tout d'abord, on peut noter que la plupart des approches manipulant des passages obtiennent des résultats nettement supérieurs à la recherche d'information classique (*Doc*) sur les quatre corpus utilisés, et plus particulièrement sur le corpus *ZIFF*, qui présente une grande diversité de tailles de documents, et sur le corpus *FR*, qui contient un grand nombre de très longs documents. Néanmoins, lors de l'utilisation de la mesure *Cosine*, on remarque les très mauvaises performances de l'approche par segments thématiques qui obtient bien souvent de moins bons résultats que cette approche classique qui considère les documents de manière globale. Par ailleurs, avec cette même mesure, on observe la supériorité des approches basées sur des séquences d'unités. Le fait que les approches utilisant des séquences de termes (T_N et RT_N) obtiennent de meilleurs résultats que celles s'appuyant sur des séquences de mots (M_N et RM_N) souligne les effets négatifs des tendances de la mesure *Cosine* à favoriser telle ou telle taille de document selon le nombre de termes contenus par la requête. Enfin, on peut noter que l'observation faite précédemment, concernant le fait que les tendances de score de la mesure *Cosine* soient mieux corrélées avec le nombre de

17. La méthode de segmentation utilisée pour l'obtention des segments thématiques considérés est la méthode *SegGen*.

termes distincts des textes qu'avec leur nombre total de termes, est vérifiée dans ce nouveau contexte d'expérimentation et a des répercussions sur les performances des systèmes puisque ce sont les approches basées sur des séquences de termes distincts (U_N et RU_N) qui obtiennent ici les meilleurs résultats. De ces premières observations avec la mesure *Cosine*, on peut retenir que le degré d'hétérogénéité des tailles des textes manipulés a un fort impact sur l'efficacité des approches.

L'utilisation de la mesure *Pivoted Cosine* semble améliorer significativement les performances de l'approche classique (ce qui correspond aux observations du tableau 6.1). L'approche utilisant des segments thématiques bénéficie quelque peu des apports de cette mesure mais le gain réalisé est moins net. Cela est dû au fait que les segments thématiques sont supposés aborder une thématique unique et donc les plus longs segments ne tendent pas nécessairement à être plus souvent pertinents. Les segments les plus longs étant plus susceptibles de provenir de longs documents, le fait de favoriser les longs segments a alors des effets positifs sur les performances de l'approche considérant des passages thématiques mais ceux-ci restent légers.

La mesure de *Cosine* normalisé par régression statistique ($NCosine$) semble améliorer les performances de la plupart des approches. Le fait que les résultats des approches considérant des séquences de mots rattrapent ceux des approches utilisant des séquences de termes (distincts ou non) indique que les biais relatifs aux variations de longueur des textes manipulés sont considérablement réduits par l'utilisation de cette mesure. L'approche utilisant des segments thématiques parait alors obtenir de bien meilleurs résultats : contrairement à ce qui a été observé dans [Kaszkiel and Zobel, 2001], cette approche obtient ici des résultats nettement supérieurs à ceux obtenus par les approches manipulant des séquences d'unités non recouvrantes (M_N, T_N et U_N). De plus, un test de Student a montré non significatives les différences de résultats entre cette approche considérant des segments thématiques et celles utilisant des séquences d'unités recouvrantes. Par ailleurs, ces dernières approches, du fait du très grand nombre de passages qu'elles manipulent (une moyenne de 115 passages par document sur le corpus FR, contre 2.5 pour les approches considérant des segments thématiques), sont peu utilisables par les systèmes de recherche d'information puisqu'elles impliquent des temps de calculs très longs et des ressources mémoire considérables (puisque l'index utilisé peut prendre des proportions gigantesques). Les segments thématiques nous paraissent alors finalement tout à fait adaptés pour une utilisation dans des approches de type *Passage Retrieval*[18]. D'autant plus que la mise en œuvre de méthodes de segmentation plus performantes, tel que l'on peut l'escompter pour les années à venir, permettra très certainement d'améliorer encore les performances des approches fondées sur les segments thématiques des documents.

18. Notons par ailleurs que, sortis de leur contexte, les segments thématiques des documents sont bien plus facilement interprétables que des séquences d'unités de taille fixe [Callan, 1994]. Une présentation des seuls passages potentiellement pertinents (hors du document les contenant) paraît alors plus facilement envisageable dans le cas des approches thématiques que dans celui des approches manipulant des séquences d'unités arbitraires.

6.4 Conclusion

De nombreux travaux ont montré qu'un découpage des documents pouvait permettre d'améliorer significativement la qualité des résultats retournés par les systèmes de recherche d'information. Néanmoins, selon les expérimentations réalisées dans [Kaszkiel and Zobel, 2001], il apparaît que le fait de travailler avec des séquences de mots de longueur arbitraire s'avère plus performant que de manipuler des passages issus d'un processus de segmentation thématique, qui sont pourtant censés mieux représenter les différentes notions abordées par les documents. Étant convaincus que le potentiel des approches de recherche d'information s'appuyant sur un découpage thématique des documents n'a pas été évalué à sa juste valeur, nous avons alors cherché à mettre en place une mesure d'estimation de la pertinence des textes qui puisse passer outre les biais relatifs aux variations de longueur des éléments manipulés dont semblent souffrir ces approches. Dans un premier temps, nous avons étudié l'influence que pouvait avoir la longueur des textes sur les scores de pertinence attribués par la mesure *Cosine*. Cette étude ayant montré de réelles tendances à favoriser certaines tailles de textes par rapport à d'autres selon le nombre de termes contenus par la requête, nous avons cherché à modéliser l'espérance des scores attribués en appliquant des techniques de régression statistique sur les données observées. La fonction finalement obtenue est alors utilisée pour normaliser les scores attribués par la mesure *Cosine* et ainsi estimer la pertinence des textes de manière plus équitable (*i.e.,* sans favoriser une taille de texte par rapport à une autre). Des expérimentations réalisées dans le cadre de la recherche documentaire classique ont tout d'abord permis de mettre en avant les bénéfices résultant de l'utilisation d'une telle mesure. Enfin, cette mesure nous a permis de montrer que la considération de segments thématiques pouvait s'avérer bien plus utile à la localisation des informations pertinentes que la prise en compte d'autres types de passages, tels que des séquences de termes de taille fixe, qui ne reflètent pas les différentes thématiques des documents.

Chapitre 7

Segmenter pour regrouper les documents pertinents

L'objectif principal de ce chapitre est d'estimer l'impact que peut avoir la segmentation thématique des documents sur les processus de clustering appliqués à la recherche d'information. Utilisant alors des similarités entre segments qui peuvent être sujettes aux mêmes difficultés relatives aux variations de longueur qu'observées avec les mesures d'estimation de pertinence, nous commençons par adapter la mesure que nous avons établie au chapitre précédent pour lui permettre de réaliser des estimations de similarités inter-textes (documents ou passages) équitables. Enfin, nous expérimentons plusieurs approches de clustering s'appuyant sur des similarités entre segments plutôt qu'entre documents entiers afin de mesurer les bénéfices que l'on peut retirer de la prise en compte de ce genre d'entité.

Sommaire

7.1 Mesures de proximité thématique et longueur des textes . . . 172
7.2 Vers des groupes plus représentatifs des sujets abordés 174
7.2.1 Approches proposées 174
7.2.2 Comparaison des approches 176
7.3 Conclusion 181

7.1 Mesures de proximité thématique et longueur des textes

Au chapitre précédent, nous avons proposé une normalisation des scores de pertinence attribués par la mesure *Cosine*. Nous nous intéressons ici aux tendances de la mesure lors de calculs de similarité entre documents (plutôt qu'entre documents et requêtes). Les poids utilisés par la mesure pour évaluer la proximité thématique de deux documents (voir formule 2.5) n'étant pas les mêmes que pour réaliser des estimations de pertinence, nous reproduisons les expérimentations réalisées en section 6.2 (création de groupes de documents selon leur taille, production de requêtes artificielles [1], ...) pour observer les tendances de la mesure *Cosine* lorsque ces poids sont utilisés. La figure 7.1 présente les distributions de similarité obtenues sur le cor-

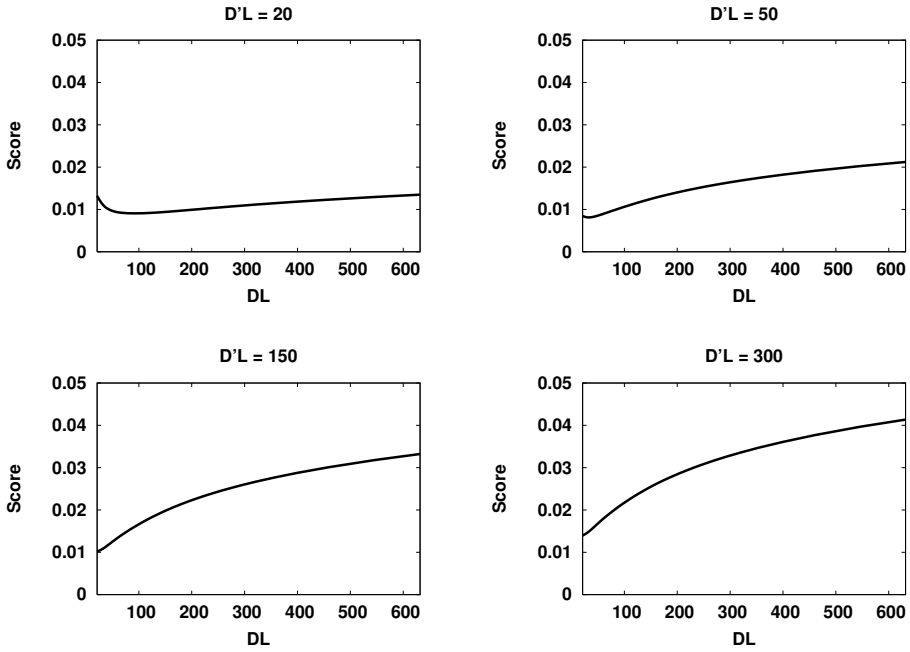

FIGURE 7.1 – Similarités entre documents

pus $ZIFF$ pour 4 tailles D'_L de documents artificiellement produits. On y remarque la claire tendance de la mesure *Cosine* à attribuer des scores de similarité plus élevés lorsque le nombre de termes contenus par les documents comparés augmente. L'application de techniques de régression statistique sur les données observées nous a

1. Puisque nous cherchons ici à modéliser l'espérance de similarité entre documents, les textes artificiellement produits sont plutôt considérés comme des documents que des requêtes. Cela revient en fait au même (ce sont dans les deux cas des concaténations de termes aléatoirement choisis dans un même document) mais nous noterons alors la longueur de ces textes D'_L (plutôt que R_L). 1000 documents "artificiels" ont été ainsi produits pour chaque taille D'_L entre 1 et 400 termes uniques (variant par paliers de 5), en choisissant chaque terme dans un document de plus de 400 termes uniques.

conduit à l'obtention de l'équation d'estimation de l'espérance de similarité suivante (avec D_L et D'_L les nombres de termes uniques des deux documents comparés) :

$$\begin{aligned} E(D_L, D'_L) = \quad & (0.68296 + 0.13079 \times \ln(D'_L) - 0.70593 \times \ln(\ln(D'_L) + 1)) \\ & + \ln(D_L) \times \\ & (0.05654 + 0.0164 \times \ln(D'_L) - 0.06364 \times \ln(\ln(D'_L) + 1)) \\ & + \ln(\ln(D_L) + 1) \times \\ & (-0.50831 - 0.10729 \times \ln(D'_L) + 0.52842 \times \ln(\ln(D'_L) + 1)) \end{aligned} \tag{7.1}$$

Le coefficient de corrélation $R^2_{y,\hat{y}}$ entre valeurs estimées par la formule 7.1 et valeurs observées tourne autour de 0.96 (avec un minimum de 0.94 pour le corpus FR). Notre modèle parait alors réaliser une relativement bonne estimation de l'espérance des scores de similarités entre documents. Elle peut donc être utilisée pour normaliser les scores de similarité obtenus par la mesure *Cosine*. Néanmoins, du fait des imperfections du modèle, on a $E(D_L, D'_L) \neq E(D'_L, D_L)$, ce qui, en appliquant la formule de normalisation 6.2, nous conduit à obtenir une relation de proximité asymétrique [2]. Afin d'établir une mesure pouvant être utilisée par la majorité des algorithmes de clustering, la normalisation des scores dépend alors des deux espérances $E(D_i, D_j)$ et $E(D_j, D_i)$:

$$NSim(D, D') = \frac{1}{2} \times \Big(1 + Sim(D, D') - \frac{E(D_L, D'_L) + E(D'_L, D_L)}{2}\Big) \tag{7.2}$$

Afin de mesurer les apports de notre mesure, nous avons, tel que c'est l'usage lors de l'établissement d'une nouvelle mesure de similarité qui est destinée à être utilisée par des systèmes de recherche d'information réalisant une catégorisation des résultats, étudié le potentiel de validité de la *Cluster Hypothesis* sur la matrice de similarités que la mesure engendre. Ce potentiel est évalué par le test des plus proches voisins (*Nearest Neighbour Test*) proposé dans [Voorhees, 1986] (voir section 3.3.2). Le tableau 7.1 présente alors les résultats de ce test [3] (où 5 voisins de chaque document pertinent sont considérés) sur les quatre corpus présentés en section 3.2, pour la mesure *Cosine* et notre mesure de *Cosine* normalisé par régression statistique. Les résultats montrent un clair rapprochement des documents pertinents, toutes les valeurs obtenues par notre mesure étant bien supérieures à celles obtenues par la mesure *Cosine*. L'ensemble des expérimentations reportées dans ce chapitre (et les suivants) utilisent alors cette nouvelle mesure qui vise à donner la même espérance de similarité à tous les couples de textes quelles que soient leurs tailles.

2. Notons que Tversky [Tversky, 1977] s'est interrogé sur l'utilité de respecter l'axiome de symétrie pour les relations de similarité, affirmant que les relations entre objets de la vie de tous les jours sont souvent asymétriques. L'exemple qui est donné concerne la relation de ressemblance entre un portrait et le visage d'une personne, on comparera de façon plus naturelle un portrait au visage d'une personne que le visage d'une personne à un portrait.

3. Notons que les résultats reportés ici ne concernent que les 50 premiers documents retournés par un système initial utilisant la mesure $NCosine$ (présentée au chapitre précédent). Notons également que seules sont considérées les requêtes pour lesquelles au moins un document pertinent appartient à cet ensemble de 50 documents.

Clustering	FR	ZIFF	AP	WSJ
Cosine	2.23	2.17	3.12	3.18
Cosine normalisé	2.90	2.43	3.25	3.44

TABLE 7.1 – Impacts de la normalisation sur les similarités entre documents

7.2 Vers des groupes plus représentatifs des sujets abordés

Dans la plupart des travaux portant sur la catégorisation des résultats d'un système de recherche d'information, les estimations de similarité entre documents sont réalisées en considérant l'ensemble des termes des documents comparés [Tombros, 2002]. Or, si, tel que nous l'avons vu au chapitre précédent, l'estimation de la similarité entre un document et une requête tire profit de la considération des passages du document, nous pensons que cela peut aussi bénéficier à l'estimation des similarités entre documents et permettre ainsi une meilleure catégorisation des résultats présentés à l'utilisateur. En effet, l'hypothèse stipulant qu'un document peut aborder diverses thématiques distinctes peut également avoir des impacts sur les jugements de proximité entre documents : malgré des développements thématiques différents, deux documents peuvent être fortement corrélés sur des zones de texte données et une mesure classique, prenant en considération l'ensemble des termes des documents comparés, risque de ne pas leur attribuer un score de similarité très élevé (étant donné que la majeure partie de leur texte traite de sujets différents). Néanmoins, si ces deux documents abordent tous deux les mêmes notions dans une partie de leur texte, il semble naturel que cela puisse avoir des répercussions sur le score de similarité qu'ils obtiennent. La considération individuelle des différents passages des documents peut alors permettre de faire ressortir des ressemblances locales. On peut alors faire l'hypothèse qu'une méthode de clustering qui se baserait sur un calcul de similarité prenant en compte les différents passages des documents conduirait à la production de groupes mieux centrés autour des différentes thématiques abordées. C'est alors ce que nous nous employons à vérifier dans cette section. Après avoir détaillé les différentes approches envisagées, nous présentons les résultats obtenus lors des expérimentations réalisées.

7.2.1 Approches proposées

De nombreuses approches peuvent être envisagées pour prendre en compte des similarités entre segments plutôt qu'entre documents entiers. En effet, alors qu'une similarité globale calculée entre deux documents correspond à un score unique, la multiplicité des segments contenus par chaque document peut conduire à l'obtention de très nombreux scores de similarité. Il faut alors définir un mode de prise en compte de ces différents scores afin de les rendre utilisables par un processus de clustering

classique (voir chapitre 2).

Une première proposition consiste à considérer comme score de similarité entre deux documents la similarité de leurs segments les plus proches :

$$Sim(D_1, D_2) = \max_{(s_i,s_j)\in\mathcal{S}(D_1)\times\mathcal{S}(D_2)} Sim(s_i, s_j) \tag{7.3}$$

où $\mathcal{S}(D_i)$ correspond à l'ensemble des segments du document D_i. Cette approche, jugeant des relations entre documents sur leurs parties communes, permet d'augmenter le score de similarité des documents abordant des sujets fortement corrélés.

Néanmoins, tant dans le cas d'un clustering classique des documents que dans celui d'un clustering ne prenant en compte que leurs passages les plus proches, l'ajout d'un document hétérogène à un cluster risque de faire dévier sensiblement celui-ci de son thème principal, puisque certaines parties de son texte peuvent en être totalement déconnectées. Des documents fortement éloignés du thème initial du cluster peuvent alors être attirés par "transitivité" (les documents hétérogènes établissent une sorte de pont entre thèmes distincts), ce qui peut être nuisible à la qualité des groupes obtenus en fin de processus. Cette seconde observation nous a conduit à proposer une approche tenant compte de la requête de l'utilisateur. Dans cette approche, la proximité d'un document avec les autres dépend uniquement de son passage le plus proche de la requête :

$$Sim(D_1, D_2) = Sim\Big(\operatorname*{argmax}_{s_i\in\mathcal{S}(D_1)}\big(Sim(s_i, R)\big), \operatorname*{argmax}_{s_j\in\mathcal{S}(D_2)}\big(Sim(s_j, R)\big)\Big) \tag{7.4}$$

où $\operatorname{argmax}_{s_i\in\mathcal{S}(D_j)}\big(Sim(s_i, R)\big)$ retourne le segment s_i, appartenant au document D_j, le plus proche de la requête (présentant le score $Sim(s_i, R)$ le plus élevé). Dans ce processus plus statique, le passage considéré dans chaque document est le même pour toutes les estimations de similarité. Les biais induits par l'ajout d'un document hétérogène à un cluster sont alors écartés. Néanmoins, une telle approche risque d'ignorer un certain nombre d'aspects des documents qui peuvent s'avérer importants. En effet, les bénéfices résultant de la catégorisation des documents pour leur présentation à l'utilisateur proviennent du fait que l'identification des documents pertinents n'est plus basée sur les seuls termes de la requête mais tire profit des relations existant entre les documents considérés. Par exemple, un document contenant un faible nombre de termes en commun avec la requête peut quand même être identifié comme pertinent grâce sa proximité à d'autres documents pertinents. Notre approche, qui ne considère, dans chaque document, que le passage le plus proche de la requête, risque d'ignorer une partie d'un document pertinent qui aurait permis de l'identifier comme tel (*i.e.,* de le grouper avec les autres documents pertinents).

Au vu de ces observations, l'idée principale est de réaliser une catégorisation préliminaire des segments thématiques pour produire des clusters plus représentatifs des thématiques abordées par les documents. La *Cluster Hypothesis* est alors étendue aux passages : les passages contenant de l'information pertinente tendent à être plus

similaires les uns avec les autres qu'avec les passages dont le thème est éloigné de la requête de l'utilisateur. Cette catégorisation préliminaire permet aux groupes de documents, finalement déduits des clusters de leurs segments thématiques, d'être mieux centrés autour de thèmes spécifiques, ce qui augmente le potentiel de validité de la *Cluster Hypothesis*.

À partir des clusters de segments thématiques des documents retournés par le système de recherche d'information initial, deux types de clusters de documents peuvent être déduits : la catégorisation peut être de type *Soft* ou *Hard* (avec ou sans recouvrement, voir section 2.1). Récemment, les algorithmes de clustering de type *Soft* [Mendes and Sacks, 2003] ont été largement étudiés dans le contexte de la catégorisation documentaire. Toutefois, à notre connaissance, aucun travail n'a tenté de tirer profit d'une segmentation thématique des documents. En outre, aucune approche n'a conduit à une catégorisation hiérarchique *Soft* des documents. Dans notre cas, le passage d'une catégorisation des segments thématiques à une catégorisation *Soft* des documents est immédiate : il suffit de remplacer chaque segment par le document dont il est tiré (si le même document apparaît plusieurs fois dans un même cluster, un unique exemplaire est conservé). Dans la hiérarchie produite, chaque document a donc été assigné aux clusters représentant les différentes thématiques de ses segments.

Pour ce qui est des algorithmes de clustering de type *Hard*, très peu de travaux ont cherché à extraire une catégorisation de documents d'une catégorisation préliminaire de fragments de documents. Dans [Conrad *et al.*, 2005], les auteurs ont expérimenté l'utilisation d'une catégorisation de fragments de textes de loi. Cette application, très spécifique, ne requiert pas l'utilisation d'une méthode de segmentation thématique puisque les textes peuvent être découpés selon les différentes dispositions juridiques qu'ils contiennent. D'autres approches extraient des mots-clés afin de créer un hiérarchie de thèmes dans laquelle les documents peuvent être assignés [Chang and Hsu, 2005]. Cependant, dans aucune de ces approches, l'impact en recherche d'information, et tout particulièrement sur la validité de l'hypothèse de clustering, n'a été évalué. Dans notre approche, une catégorisation des documents de type *Hard* peut être déduite de la catégorisation des segments thématiques en assignant chaque document au cluster qui lui est le plus proche (en calculant la moyenne des distances de chaque segment d'un cluster donné avec le document en question).

7.2.2 Comparaison des approches

Les expérimentations de nos approches se font sur les quatre corpus présentés en section 3.2. Néanmoins, dans le but de réaliser des tests sur des documents fortement hétérogènes, un corpus additionnel, le corpus ART, a été constitué artificiellement par concaténation, pour chaque document, de quatre articles du corpus AP. Sur ce corpus, aucune méthode de segmentation n'est employée, les frontières thématiques correspondent aux séparations entre articles. Ce corpus ART ayant été construit

artificiellement, il n'existe pas de liste de jugements de pertinence. Nous considérons alors un document du corpus *ART* comme pertinent si il contient au moins un document pertinent de *AP*.

Corpus	ART	ZIFF	AP	WSJ	FR
Segments thématiques	84510	138998	156147	179521	101326
Requêtes avec au moins un pertinent	34	31	33	39	13
Documents pertinents par requête	45.09	82.87	43.24	61.69	29
MAP de la recherche initiale	0.353	0.397	0.381	0.478	0.494

TABLE 7.2 – Statistiques des corpus utilisés

Par ailleurs, les résultats présentés ci-après ne réalisent la catégorisation, par la méthode de clustering *Group-Average* [4] (cette méthode ayant montré les meilleurs résultats dans la plupart des études, voir par exemple [Tombros *et al.*, 2002]), que des 50 premiers documents (ce nombre de documents ayant montré de bons résultats dans [Tombros *et al.*, 2002]) retournés par une recherche initiale (utilisant la mesure *NCosine*) en réponse à une requête courte (*Title*). Afin de ne pas distordre les résultats par obtention de valeurs nulles pour certaines requêtes, nous avons choisi de ne prendre en compte que les requêtes possédant au moins un document pertinent dans les 50 documents considérés [5].

Les expérimentations réalisées comparent les approches suivantes :

- D : Clustering classique des documents ;
- C_M : Clustering des documents où la similarité entre deux documents correspond à la similarité de leurs passages les plus proches (formule 7.3) ;
- C_R : Clustering des documents où la similarité d'un document avec les autres dépend des termes de son passage le plus proche de la requête (formule 7.4) ;
- S : Clustering *Soft* des documents déduit d'un clustering préliminaire des segments thématiques ;
- H : Clustering *Hard* des documents déduit d'un clustering préliminaire des segments thématiques.

L'ensemble des clusterings ont été réalisés en utilisant des similarités inter-textes calculées selon la mesure *NCosine* définie en section 7.1. Néanmoins, dans le cas des catégorisations de segments thématiques (approches S et H), il est fort probable que les segments d'un même document aient été rédigés par une même personne, utilisant son style et son vocabulaire propre, et donc aient un certain nombre d'as-

4. Afin de pouvoir évaluer nos approches sans avoir à définir un nombre de groupes *a priori*, nous nous plaçons dans un contexte de clustering hiérarchique (ce qui nous permet par ailleurs d'appliquer la mesure $MK1$ de recherche du groupe optimal présentée en section 3.3.2).

5. Des statistiques additionnelles à celles présentées par le tableau 3.1 sont données dans la table 7.2. Dans cette table, le nombre de segments est calculé sur l'ensemble des documents du corpus. Les statistiques MAP (précision moyenne) et "Requêtes avec au moins un pertinent" ne prennent en compte que les documents pertinents retournés parmis les 50 premiers résultats du système initial.

Clustering	ART	FR	ZIFF	AP	WSJ
D	2.57	2.90	2.43	3.25	3.44
C_M	3.30	3.15	3.16	3.20	3.66
C_R	3.24	3.08	3.12	3.23	3.60

TABLE 7.3 – Test des plus proches voisins

pects en commun. Ces segments ont alors de grandes chances d'obtenir des scores de similarité très élevés et donc de se retrouver dans les mêmes clusters, ce qui revient à peu de choses près à réaliser une catégorisation classique de documents (approche D). Afin d'écarter ce biais, la similarité entre deux segments appartenant au même document est fixée arbitrairement à la similarité moyenne entre segments de documents différents.

Propension à regrouper l'information pertinente

Afin de mesurer les apports de la prise en compte des passages dans les calculs de similarité, nous commençons ces expérimentations par l'étude du potentiel de validité de la *Cluster Hypothesis* sur les matrices de similarités utilisées par les différentes approches. Étant donné que l'on ne dispose pas de jugements de pertinence pour les différents segments considérés, le test des plus proches voisins (*cf.,* section 3.3.2) ne peut pas être appliqué pour les approches S et H qui utilisent directement les matrices de similarités entre segments pour la production des clusters de documents. Néanmoins, l'impact de la segmentation thématique sur le potentiel de validité de la *Cluster Hypothesis* peut être évalué pour les approches C_M et C_R dans lesquelles une matrice de similarités entre documents est construite à partir des similarités entre segments. Le tableau 7.3 reporte alors les scores obtenus par le test des plus proches voisins (avec un nombre de 5 voisins considérés) avec les matrices de similarités utilisées par les approches D, C_M et C_R sur nos cinq corpus d'évaluation. Les résultats montrent que la considération des segments pour estimer les similarités entre documents permet d'augmenter le potentiel de validité de la *Cluster Hypothesis*. En effet, sur l'ensemble des corpus (sauf sur AP où la considération des segments thématiques ne permettait déjà pas d'améliorer significativement l'ordre des résultats, voir la table 6.6) les résultats font apparaître que les documents pertinents sont plus proches les uns des autres dans les matrices utilisées par les approches C_M et C_R que dans celle de l'approche D. Cela rend compte d'une meilleure capacité à regrouper les documents pertinents lorsque les segments thématiques sont considérés.

Qualité du groupe optimal

À l'instar de [Tombros *et al.*, 2002], les performances des approches de recherche d'information basées sur une catégorisation des résultats sont évaluées selon la qualité du meilleur cluster de la hiérarchie produite (mesure $MK1$, voir section 3.3.2), où la qualité d'un cluster est évaluée selon des critères de précision et de rappel[6]. Le tableau 7.4 présente les résultats des différentes approches en terme de qualité[7] (E), nombre de documents pertinents (P) et nombre total de documents (N) du meilleur cluster de la hiérarchie.

Les approches D_R et P_R reportées dans ce tableau correspondent respectivement à la liste ordonnée retournée par le système initial (*Document Retrieval*) et à une liste réordonnée selon la similarité du meilleur passage de chaque document (*Passage Retrieval*). Toutes deux sont évaluées selon le critère $MK3$, permettant un rapprochement de leurs performances avec celles des approches réalisant une catégorisation des résultats (voir section 3.3.2). Il est alors possible de remarquer que l'ensemble des approches réalisant une catégorisation des documents (D, C_M, C_R, S et H) permettent l'obtention de meilleurs résultats que les approches présentant une liste ordonnée de documents à l'utilisateur (D_R et P_R). La *Cluster Hypothesis* semble donc se vérifier sur les corpus considérés ; la catégorisation des documents retournés par le système tend à regrouper les documents pertinents dans un même cluster.

D'une manière générale, selon les résultats présentés, il semble que les observations réalisées sur les matrices de similarités (par le test des plus proches voisins) se reportent sur la qualité des groupes effectivement produits par les différentes approches expérimentées : la prise en compte des segments a des effets très bénéfiques sur le regroupement des documents pertinents. Sur chaque corpus où l'approche P_R obtient des résultats significativement supérieurs à ceux de D_R (*i.e.,* sur ART, FR, WSJ et $ZIFF$), il apparaît en effet que la prise en compte des segments thématiques dans le processus de catégorisation des documents conduit à l'obtention de groupes de meilleure qualité. L'hypothèse principale de ce chapitre semble donc vérifiée (du moins sur les corpus utilisés), l'individualisation des segments thématiques des documents a un impact très positif sur les groupes de documents présentés à l'utilisateur.

Sur le corpus ART, les approches C_M et C_R obtiennent les meilleurs résultats. Dans cette collection, les passages d'un même document étant des articles extraits d'un journal généraliste, il est probable que chaque document ne possède qu'un seul passage contenant les termes de la requête. En conséquence, la meilleure similarité entre passages de documents différents est enregistrée entre segments ayant trait à l'information recherchée par l'utilisateur. Le regroupement des documents pertinents est alors facilité par la non prise en compte des autres passages, proba-

6. Tel que réalisé dans [Tombros *et al.*, 2002], le rappel considère l'ensemble des documents pertinents existant dans le corpus plutôt que les seuls documents pertinents retournés par le système initial.

7. Le score de la mesure utilisée est égal à 0 lorsque tous les documents pertinents du corpus appartiennent au cluster concerné (*Rappel=1*) et que ce cluster ne contient aucun document non pertinent (*Precision=1*). Ainsi, plus ce score est faible, meilleure est la qualité du cluster concerné.

Group Average		$\beta = 0.5$			$\beta = 1$			$\beta = 2$		
		E	*P*	*N*	*E*	*P*	*N*	*E*	*P*	*N*
ART	D_R	0.716	6.618	22.06	0.745	8.588	33.06	0.728	9.088	38.32
	P_R	0.631	6.765	16.29	0.684	8.735	25.50	0.682	9.088	29.59
	D	0.594	6.941	14.71	0.687	8.441	22.82	0.702	9.000	29.65
	C_M	**0.528**	7.206	12.03	**0.621**	8.794	19.50	**0.643**	9.147	25.97
	C_R	**0.528**	7.206	12.03	**0.621**	8.794	19.50	**0.643**	9.147	25.97
	S	0.532	7.235	12.24	0.627	8.824	20.32	**0.643**	9.088	24.76
	H	0.530	7.176	12.18	0.629	8.647	19.41	0.651	9.000	26.00
FR	D_R	0.645	3.923	11.00	0.695	4.000	11.23	0.692	4.154	12.69
	P_R	0.602	3.923	9.462	0.638	4.077	10.85	0.643	4.154	11.46
	D	0.484	3.846	5.308	0.570	4.000	5.615	0.607	4.077	6.462
	C_M	0.463	3.846	4.769	0.561	4.000	5.231	0.602	4.077	6.000
	C_R	0.470	3.846	4.923	0.565	4.000	5.385	0.603	4.077	6.154
	S	0.417	3.923	4.538	0.526	4.077	5.000	**0.571**	4.154	5.923
	H	**0.413**	3.846	4.308	**0.525**	4.077	4.923	0.583	4.077	4.923
ZIFF	D_R	0.688	10.16	30.32	0.730	10.77	33.81	0.728	11.03	35.87
	P_R	0.659	10.16	26.19	0.708	10.87	31.52	0.716	11.13	33.71
	D	0.600	9.742	22.10	0.684	10.71	27.65	0.704	11.10	31.81
	C_M	0.588	10.16	21.23	0.678	10.94	26.19	0.702	11.13	31.94
	C_R	0.589	10.16	21.35	0.676	11.03	26.65	0.700	11.16	32.29
	S	0.575	10.16	20.74	**0.661**	11.42	25.81	**0.688**	12.03	31.19
	H	**0.569**	10.06	18.35	0.662	10.87	23.32	0.695	11.29	31.55
AP	D_R	0.622	6.970	16.91	0.673	8.424	23.21	0.679	9.576	31.52
	P_R	0.613	7.030	16.45	0.671	8.424	21.33	0.680	9.576	31.91
	D	0.526	7.333	12.12	0.618	8.848	19.42	0.631	9.667	26.61
	C_M	0.525	7.333	12.06	0.618	8.788	19.09	0.632	9.667	27.21
	C_R	0.526	7.303	12.09	0.618	8.909	19.61	0.630	9.697	28.58
	S	0.524	7.394	12.18	**0.616**	9.152	20.67	**0.629**	9.727	28.88
	H	**0.523**	7.303	11.79	**0.616**	8.909	18.94	0.630	9.697	28.30
WSJ	D_R	0.679	11.10	28.08	0.728	12.08	33.46	0.730	12.33	37.38
	P_R	0.643	11.08	24.10	0.702	12.13	30.74	0.707	12.49	33.33
	D	0.593	10.95	19.13	0.680	12.05	23.33	0.711	12.28	27.44
	C_M	0.584	10.87	18.33	0.670	12.21	23.62	0.704	12.46	28.72
	C_R	0.586	10.90	18.67	0.668	12.26	24.74	0.701	12.51	29.90
	S	0.563	11.21	17.46	**0.651**	13.00	24.69	**0.692**	13.38	28.59
	H	**0.561**	11.10	17.13	0.654	12.49	22.70	0.698	12.87	26.31

TABLE 7.4 – Performances des approches : qualité du groupe optimal

blement étrangers à la requête de l'utilisateur. Cependant, dans les autres corpus qui contiennent des textes plus homogènes, certains documents peuvent se ressembler sur des passages fortement déconnectés du besoin de l'utilisateur. L'approche C_M est alors susceptible de considérer des similarités entre passages n'ayant que peu d'intérêt pour la recherche d'information réalisée, et ainsi de créer des groupes de documents aux thématiques centrales éloignées du sujet de l'utilisateur. L'approche C_R, quant à elle, n'obtient pas de meilleurs résultats sur ces corpus : le fait de ne considérer que le meilleur passage (le plus proche de la requête) de chaque document conduit à ignorer un certain nombre d'aspects qui peuvent s'avérer utiles pour regrouper l'ensemble des documents pertinents.

Les approches dépendant d'une catégorisation préliminaire des segments thématiques (S et H) ne présentent pas ces différents biais et obtiennent alors les meilleurs résultats sur ces corpus de textes réels. Un test de Student a montré que les différences entre les résultats obtenus par ces approches et ceux obtenus par les approches classiques sont statistiquement significatives sur les corpus FR, $ZIFF$ et WSJ. Enfin, l'approche réalisant une catégorisation *Soft* des documents semble obtenir des résultats légèrement supérieurs à ceux présentés par l'approche produisant une catégorisation de type *Hard*. Cela est dû à la difficulté d'affectation des documents aux clusters : l'algorithme doit choisir un aspect du document parmi les thèmes représentés par les clusters contenant ses segments thématiques. Néanmoins, l'approche utilisant une catégorisation de type *Hard* conduit à une meilleure précision du cluster contenant les documents pertinents.

7.3 Conclusion

Dans ce chapitre, nous avons expérimenté l'impact que pouvait avoir l'utilisation d'un processus de segmentation thématique des documents sur la catégorisation des résultats d'une recherche d'information. L'objectif était alors de considérer individuellement les différentes thématiques de chaque document, dans le but de produire des groupes de documents mieux centrés autour de sujets donnés. Même si les similarités intra et inter-clusters n'ont pas été nécessairement améliorées par la prise en compte de tels passages, les groupes de documents obtenus semblent être plus représentatifs des thèmes abordés par les documents. Les expériences réalisées ont montré qu'un découpage thématique des documents permettait d'obtenir des clusters présentant une meilleure proportion de documents pertinents, et ainsi d'aider l'utilisateur dans sa tâche de localisation des informations correspondant à ses besoins.

Troisième partie

Organisation de l'information pertinente

Les unités de texte que nous serons amenés à manipuler pour composer notre document final étant maintenant clairement définies, il reste à déterminer la manière de procéder pour sélectionner les éléments devant y figurer. L'objectif fixé étant la production d'une liste de segments représentant un aperçu de l'ensemble des informations qu'il est possible de trouver dans un corpus en rapport avec l'expression d'un besoin d'information, la mise en place d'un processus permettant la mise en évidence des différentes thématiques en jeu est alors nécessaire. L'utilisation de techniques de clustering nous apparaît alors tout à fait adaptée : puisque l'objectif est de faire émerger les différents aspects d'un sujet, il semble pertinent de catégoriser les segments des documents retournés par une recherche préliminaire pour identifier des groupes thématiques desquels il sera possible d'extraire les segments devant figurer dans notre document composite final. Néanmoins, tel que nous le verrons dans un premier chapitre, la tendance naturelle des textes pertinents à se regrouper dans un même cluster peut se poser en obstacle à l'émergence de la pluralité des aspects du besoin exprimé par l'utilisateur. La mise en place de nouvelles méthodologies d'évaluation nous permet alors de mettre en évidence qu'une prise en compte du contexte dans lequel le clustering est réalisé peut conduire à l'obtention d'un meilleur partitionnement de l'information pertinente. Le second chapitre de cette partie s'emploie alors à l'établissement d'un algorithme de clustering orienté requête qui cherche à organiser les clusters autour du besoin d'information exprimé par l'utilisateur, rendant alors possible la production d'une liste de représentants de clusters constituant un réel aperçu des différents aspects d'un sujet. L'application d'un tel processus de clustering au niveau des segments thématiques des documents nous permet finalement de proposer un document composite visant à décrire le plus succintement possible les principales informations qu'un utilisateur pourra trouver dans un corpus en rapport avec le besoin qu'il a exprimé.

Chapitre 8

Concentration vs. Distribution de l'information pertinente

Nous nous interrogeons ici sur le meilleur moyen d'aider un utilisateur dans sa recherche d'information : vaut-il mieux, à l'instar de la plupart des systèmes, chercher à concentrer l'ensemble des documents pertinents dans un seul et même cluster, ou bien plutôt tenter de distribuer l'information pertinente sur l'ensemble des groupes afin d'en faire émerger la pluralité des aspects ? L'objectif est alors de définir des méthodologies d'évaluation permettant la comparaison de ces deux points de vue opposés, afin d'identifier le mode de présentation des résultats permettant la meilleure localisation des informations recherchées. Nous terminons ce chapitre par une comparaison entre différents systèmes basée sur ces nouvelles mesures d'évaluation mises en place.

Sommaire

8.1 Remise en cause de la Cluster Hypothesis 186
8.2 Évaluer l'accès à l'information 189
8.2.1 Parcours optimal 192
8.2.2 Parcours moyen 195
8.2.3 Parcours orienté par la pertinence des documents 197
8.2.4 Parcours orienté par la proximité des documents pertinents . . . 198
8.2.5 Étude des mesures proposées 201
8.3 Comparaison des systèmes 207
8.4 Conclusion 210

8.1 Remise en cause de la Cluster Hypothesis

Bien que la validité de la *Cluster Hypothesis* ait été vérifiée à maintes reprises [Van Rijsbergen, 1979; Cutting *et al.*, 1992; Hearst and Pedersen, 1996], un certain nombre de travaux émettent de sérieux doutes quant à la capacité à diviser l'espace des similarités en deux zones bien distinctes, comprenant l'ensemble des documents pertinents d'un côté et celui des non pertinents d'un autre. Pour Salton et Buckley [Salton and Buckley, 1994], l'hypothèse selon laquelle la totalité des documents pertinents à une requête puissent être regroupés dans une zone particulière de l'espace des similarités semble largement optimiste, affirmant même qu'une telle configuration ne s'observe que très rarement. Certes, les documents pertinents tendent à être plus proches les uns des autres que des non pertinents mais cela reste une tendance et il est difficile d'affirmer que des documents non pertinents ne puissent pas se glisser dans la zone des documents pertinents (et réciproquement). En raison d'un style d'écriture particulier, par exemple, le vocabulaire d'un document pertinent peut ne posséder que de faibles relations de similarité avec les autres documents pertinents du corpus. Selon [Bellot, 2000], "d'un côté l'existence d'homonymes fait que deux documents peuvent sembler proches tout en traitant de thématiques différentes, et de l'autre, l'existence de synonymes peut faire paraître éloignés deux documents pourtant proches thématiquement". D'après des expérimentations décrites dans [de Loupy *et al.*, 1999] ou [Marteau *et al.*, 1999], les documents pertinents sont plus généralement répartis dans plusieurs zones de l'espace, il est alors difficile de trouver une unique séparation efficace entre les différents documents du corpus. Non seulement les différences de vocabulaire peuvent conduire à affaiblir le niveau de similarité entre certains documents pertinents, mais il est également possible que l'ensemble des documents pertinents abordent la question sous différents points de vue, dans différents contextes ou même, traitent de différents aspects, bien distincts, du sujet qui intéresse l'utilisateur. L'obtention d'un groupe qui puisse à la fois rassembler la totalité des documents pertinents tout en excluant l'ensemble des non pertinents est alors assez improbable.

Alors que la plupart des systèmes réalisant une catégorisation des résultats considèrent la *Cluster Hypothesis* comme un phénomène largement bénéfique, et s'y appuient alors pour tenter de créer le cluster le plus informatif possible, nous aurons plutôt tendance à la considérer comme un obstacle majeur à la production de groupes permettant à l'utilisateur de localiser rapidement les informations qu'il recherche. En effet, la majorité des documents pertinents tendant à se regrouper dans un seul et même cluster, $k-1$ des k clusters présentés sont supposés être largement déconnectés de la requête formulée, ce qui pose, selon nous, un certain nombre de problèmes.

Tout d'abord, la majorité des informations initialement présentées à l'écran risquent de ne pas correspondre aux besoins de l'utilisateur. Il suffit alors que le représentant du cluster contenant l'ensemble des documents pertinents n'ait pas été

judicieusement choisi pour que l'utilisateur soit incapable de localiser les informations qu'il recherche. De plus, quand bien même le représentant s'avère intéressant, l'impression première que l'utilisateur peut avoir est que peu d'informations correspondent à son sujet (ou que le système de recherche est mauvais), ce qui peut le conduire à reformuler sa requête (ou changer de système) jugeant alors que la piste suivie ne lui permettra pas de satisfaire ses besoins informationnels.

Par ailleurs, tel que nous venons de l'évoquer, la *Cluster Hypothesis* ne se vérifie pas toujours pleinement, elle ne fait qu'énoncer des tendances observées. Ainsi les plus fortes similarités entre documents pertinents ne concernent pas toujours la totalité de ces documents. De plus, les techniques de clustering employées n'étant pas toujours infaillibles, certains documents, mêmes proches des autres documents pertinents, peuvent se retrouver dans un cluster différent. Le fait qu'une majorité de documents pertinents appartiennent à un même groupe conduit les documents pertinents malencontreusement contenus dans les autres clusters à être bien isolés parmi des documents non pertinents. Ces documents isolés ont alors peu de chances de trouver dans le représentant de leur cluster (document ou liste de termes) un "porte-parole" efficace. Lorsque la majorité des documents pertinents sont contenus dans un même cluster, l'utilisateur est amené à ne s'intéresser qu'à ce cluster en particulier, pouvant alors passer à côté de documents qui auraient pu compléter sa recherche, en apportant des informations complémentaires, en faisant part d'un point de vue différent ou même, en traitant d'un aspect différent de la question. En tentant de regrouper l'ensemble des documents pertinents dans un même cluster, on prend le risque de restreindre les informations portées à la connaissance de l'utilisateur à un seul point de vue. Le paradoxe est alors considérable : alors que l'on cherche à aider un utilisateur dans sa collecte d'informations, on risque de restreindre sa perception du sujet en l'incitant à ne visiter qu'un seul cluster susceptible de ne contenir que des documents abordant la question sous un même angle.

Enfin, le fait de rassembler l'ensemble des documents pertinents dans un même groupe ne permet pas de faire émerger la structure de l'information pertinente. Or, une même requête peut comprendre un certain nombre d'aspects bien distincts. L'utilisateur, face à un jeu de clusters dans lequel un unique groupe lui est présenté comme étant susceptible de lui être utile, n'a alors aucune idée de la multitude d'aspects que son sujet peut présenter. Lorsqu'il entre dans le cluster des documents pertinents, il est alors face à une liste ordonnée de documents, certes "filtrée" mais qu'il faut tout de même parcourir linéairement jusqu'à avoir le sentiment d'avoir collecté assez d'informations. Un problème majeur se pose alors : l'utilisateur doit prendre la décision d'arrêter la collecte d'informations alors qu'il ne connait pas la

diversité des textes en relation avec sa requête[1].

Pour ces différentes raisons, nous pensons que le fait de chercher à regrouper l'ensemble des documents pertinents dans un même groupe n'est pas nécessairement le meilleur choix. Une distribution de l'information pertinente est alors certainement préférable à sa concentration dans un unique cluster. La *Cluster Hypothesis* se pose alors bien comme un obstacle à surmonter, la majorité des documents pertinents présentant une tendance naturelle à se regrouper.

Un certain nombre d'approches, les approches de clustering orienté requête [Tombros, 2002], proposent de considérer la requête de l'utilisateur dans le processus de clustering, affirmant qu'une prise en compte du contexte dans lequel la catégorisation des documents est effectuée constitue un facteur déterminant pour l'obtention d'un partitionnement adapté à l'utilisation que l'on souhaite en faire. Selon le sujet qui intéresse l'utilisateur, la catégorisation la plus adaptée d'un même ensemble de documents n'est en effet pas nécessairement la même. Ainsi, [Tombros *et al.*, 2003] proposent par exemple de ne considérer, lors du calcul de la similarité de deux documents, que leurs phrases les plus proches de la requête. Cela permet de ne s'intéresser qu'à des indices de proximité thématique correspondant au sujet qui intéresse l'utilisateur et ainsi de former des clusters susceptibles de mieux correspondre à ses attentes. Néanmoins, de la même manière que dans le cas de l'approche C_R explorée au chapitre précédent, cette approche prend le risque d'ignorer de nombreuses parties de texte qui, bien que ne contenant pas les termes de la requête, auraient pu être utiles au rapprochement de documents aux thématiques fortement connectées. D'autres approches, telles que celles présentées dans [Chang and Hsu, 1997], [Iwayama, 2000] ou [Tombros and Van Rijsbergen, 2004], proposent de modifier l'espace des similarités entre documents en incluant une prise en compte de la requête dans les calculs de similarité entre documents. La *Query Sensitive Similarity Measure* (*QSSM*), présentée dans [Tombros and Van Rijsbergen, 2004], semble être l'approche la plus performante d'entre elles. Contrairement aux autres approches, qui modifient simplement les similarités en augmentant les poids des termes de la requête dans les représentations des documents, la mesure *QSSM* réalise, pour le calcul de la similarité entre deux documents, un produit entre leur score de similarité thématique classique (en terme de *Cosine* ou *Cosine* normalisé, par exemple) et un score de proximité à la requête du vecteur correspondant à l'intersection de leurs deux représentations vectorielles (seuls les termes communs sont conservés dans ce vecteur, le poids de chaque terme du vecteur correspond alors à la moyenne de ses poids dans les vecteurs individuels des deux documents). De cette manière, les documents contenant des termes de la requête différents, ce qui peut être considéré

1. Notons néanmoins qu'un second processus de clustering peut être appliqué à ce cluster particulier, permettant ainsi de faire émerger une certaine structure de l'information pertinente. Dans ce cas cependant, le problème précédent risque d'être amplifié par le fait que l'on risque de donner à l'utilisateur l'impression qu'il se trouve face à un système lui permettant de percevoir l'ensemble des aspects de son sujet, alors que les documents abordant réellement la requête sous un angle différent sont susceptibles d'être contenus par les autres clusters et donc de n'être pas représentés par les groupes qui lui sont présentés.

comme un indicateur de thématiques différentes, obtiennent un score de similarité minoré. À l'inverse, deux documents partageant un grand nombre de termes de la requête, et qui donc ont des chances de correspondre à un même aspect du sujet, voient leur similarité renforcée par cette prise en compte de leur proximité commune à la requête. Bien que ces approches, y compris cette mesure *QSSM* que nous étudions par la suite, aient été pour la plupart conçues dans le but d'augmenter les capacités des méthodes de clustering à regrouper l'ensemble des documents pertinents dans un même cluster, elles peuvent selon nous permettre de produire des clusters mieux organisés autour de la requête de l'utilisateur et ainsi, lorsque celle-ci comporte un certain nombre d'aspects bien distincts, de distribuer l'information pertinente dans des clusters différents.

Le problème est que, bien souvent, les systèmes de recherche d'information réalisant une catégorisation des résultats sont évalués uniquement sur leur capacité à regrouper les documents pertinents dans un seul et même cluster (c'est par exemple le cas de la mesure *MK1* [Jardine and Van Rijsbergen, 1971] que nous présentons en section 3.3.2 et utilisons en section 7.2). Le fait de produire, pour une requête donnée, différents groupes de documents pertinents correspondant à différents aspects du sujet recherché n'est alors pas mis en valeur. Les reconstructions de listes présentées en section 3.3.2 se posent en alternatives aux évaluations basées sur l'application directe de la *Cluster Hypothesis*. Permettant de simuler des parcours de clusters, elles permettent d'évaluer les systèmes sur l'ensemble des clusters qu'ils produisent plutôt que sur le seul cluster contenant le plus grand nombre de documents pertinents. Néanmoins, nous pensons que les approches proposées pour reconstruire ces listes présentent un certain nombre de biais qui conduisent bien souvent à favoriser les systèmes regroupant l'ensemble des documents pertinents dans un même cluster. La prochaine section est alors dédiée à la présentation de ces biais expérimentaux et à la proposition de mesures plus équitables, visant à mettre en valeur les apports réalisés par les systèmes permettant l'identification, lorsqu'ils existent, des différents aspects de la requête.

8.2 Évaluer l'accès à l'information

Tel que présenté en section 3.3.2, afin d'évaluer les systèmes réalisant un clustering des résultats de la même façon que les systèmes classiques, certaines approches ont proposé de reconstruire des listes ordonnées de documents à partir des clusters formés [Hearst and Pedersen, 1996; Silverstein and Pedersen, 1997; Bellot and El-Bèze, 1999]. Après avoir établi un ordre entre les clusters et entre les documents à l'intérieur de chaque cluster pour considérer l'ensemble des groupes produits par le système à évaluer comme un ensemble ordonné $\mathcal{C} = \{C_1, \ldots, C_k\}$ de k listes de documents $C_i = \{C_i^1, \ldots, C_i^{|C_i|}\}$, où C_i^j correspond au j-ième document du i-ième cluster C_i, les approches définissent un parcours à travers ces différentes listes pour produire une liste ordonnée L contenant l'ensemble des n documents sur

laquelle une mesure de précision moyenne peut être appliquée (cette liste L peut alors être considérée comme un ré-ordonnancement, par utilisation de techniques de clustering, de la liste $\mathcal{D} = \{D_1, \ldots, D_n\}$ des n premiers documents retournés par le système initial).

Les deux parcours de clusters les plus classiques sont le parcours en profondeur et le parcours en largeur. Alors que le parcours en profondeur examine les clusters les uns après les autres en commençant par celui possédant le plus fort potentiel de pertinence, le parcours en largeur considère séquentiellement les premiers documents non encore examinés de chaque liste. Ces deux parcours sont souvent utilisés conjointement pour évaluer les performances des systèmes. Ils présentent néanmoins, selon nous, un certain nombre de limites. Tout d'abord, les évaluations réalisées selon ces deux types de parcours observent une corrélation inverse : alors que le parcours en profondeur favorise les systèmes regroupant l'ensemble des documents pertinents dans un même cluster, le parcours en largeur favorise les systèmes réalisant une distribution des documents pertinents sur l'ensemble des groupes. L'amélioration du score d'une de ces deux évaluations tend alors à faire diminuer l'autre, ce qui rend difficile l'interprétation des résultats obtenus. D'autant plus que l'équilibre de ces deux mesures n'est pas évident. Il est alors difficile de déterminer la dominance d'un partitionnement sur un autre lorsque l'un des deux obtient un meilleur score par un parcours en profondeur et l'autre un meilleur score par un parcours en largeur.

La considération conjointe de ces deux critères pour comparer des clusterings produits par différentes méthodes tend à pénaliser ceux qui observent une distribution de l'information pertinente sur l'ensemble des clusters puisqu'il est plus difficile d'obtenir un score élevé selon un parcours en largeur que selon un parcours en profondeur. En effet, même si les différents groupes produits par une méthode donnée sont bien représentatifs des différents aspects de la requête formulée par l'utilisateur, il se peut que certains d'entre eux ne correspondent pas à ce qu'attendaient les annotateurs lors de l'établissement des jugements de pertinence des documents. Or, l'existence d'un seul groupe ne contenant aucun document pertinent suffit à réduire considérablement le score obtenu par un parcours en largeur. Par ailleurs, le recul d'un document pertinent d'une seule position dans la liste d'un cluster (ce qui correspond à une perte mineure) implique son recul de k positions dans la liste finale représentant un parcours en largeur (alors qu'il ne perd qu'une seule place dans un parcours en profondeur).

De plus, les caractéristiques des partitionnements produits (tailles des clusters, degré de distribution de l'information pertinente sur l'ensemble des groupes) ont un fort impact sur les scores obtenus. Par exemple, les documents non pertinents situés en fin de cluster (ou de la liste représentant le cluster), qui ne devraient avoir aucun effet sur l'évaluation puisqu'ils n'ont que très peu de chance d'être examinés par l'utilisateur, ont un effet très négatif sur le score obtenu par un parcours en profondeur. Les écarts entre les nombres de documents pertinents contenus par chaque cluster ont un effet très négatif sur le score obtenu par un parcours en largeur :

si un cluster contient plus de documents pertinents que les autres, un parcours en largeur risque de considérer ses derniers documents pertinents après certains documents non pertinents des autres groupes (même si ces documents pertinents ne sont pas classés après des documents non pertinents dans leur propre groupe).

Enfin, les deux évaluations dépendent fortement de l'ordre dans lequel sont présentés les clusters (ils sont généralement ordonnés selon le potentiel de pertinence de leur document le plus proche de la requête). Cependant, alors que l'ordre dans lequel sont présentés les représentants des clusters peut avoir de l'importance (puisque l'utilisateur les examine logiquement dans l'ordre dans lequel ils lui sont présentés), cet ordre établi n'a pas d'impact direct sur la route que l'utilisateur emprunte à travers les clusters : à partir des descriptions présentées, l'utilisateur identifie les aspects qui semblent répondre au mieux à ses besoins et parcourt les clusters correspondants en priorité. Par exemple, le fait qu'un cluster soit présenté en cinquième position n'implique pas que le contenu de ce cluster sera examiné en cinquième position, tout dépend de l'intérêt que l'utilisateur lui porte (contrairement à l'ordre des documents dans les clusters que l'utilisateur est supposé suivre, puisqu'il n'y a pas de raison qu'il commence à examiner la fin de la liste présentée avant son début).

Dans [Leuski, 2001a], Leuski propose un compromis entre les deux extrêmes que représentent le parcours en profondeur et le parcours en largeur en considérant les jugements de pertinence établis lors de l'annotation du corpus. L'idée est de définir une stratégie de parcours qui tente de simuler le comportement qu'aurait pu avoir un utilisateur réel face aux différents clusters présentés par le système à évaluer : lorsque le nombre de documents non pertinents examinés dans un cluster dépasse le nombre de documents pertinents examinés dans ce même cluster, le processus stoppe le parcours de la liste correspondant au cluster courant pour s'intéresser à un autre groupe de documents. Le nouveau groupe choisi correspond alors au cluster dans lequel la meilleure proportion de documents pertinents a été observée sur les documents examinés (en cas d'égalité le cluster choisi est celui dont le premier document non encore examiné possède la meilleure estimation de pertinence). Cette stratégie, qui peut être rapprochée des processus d'expansion de requêtes par réinjection de pertinence, simule le parcours qu'un utilisateur aurait pu emprunter en ce sens qu'elle se sert des éléments examinés pour orienter son parcours vers les clusters les plus susceptibles de contenir les informations pertinentes. La liste de documents finalement obtenue par le parcours réalisé est alors supposée mieux refléter la réalité que les parcours de clusters en profondeur ou en largeur communément employés, ce qui laisse augurer d'une meilleure évaluation du système [2].

Néanmoins, si cette évaluation limite quelque peu les biais présentés par les parcours en profondeur et en largeur, elle favorise elle aussi les méthodes regroupant

2. Notons qu'en utilisant cette stratégie de parcours, Leuski a montré que les systèmes réalisant une catégorisation des résultats pouvaient s'avérer au moins aussi performants que la plupart des systèmes utilisant des retours de pertinence (*Relevance Feedbacks*) pour enrichir la requête formulée par l'utilisateur [Leuski, 2001a].

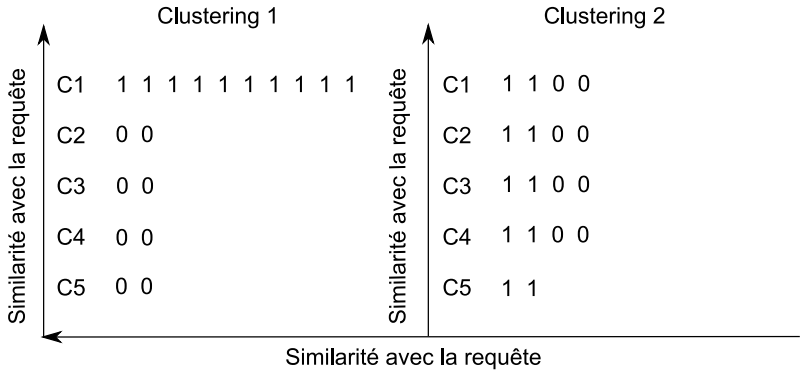

FIGURE 8.1 – Deux partitionnements de 18 documents en 5 clusters.

la plupart des documents pertinents dans un même groupe. En effet, l'intervalle existant entre le dernier examen d'un document pertinent dans un cluster et la prise de décision de changer de cluster implique des prises en compte de documents non pertinents. Or, le nombre de changements de clusters requis est plus important lors de parcours de listes correspondant à un partitionnement produit par une méthode cherchant à distribuer l'information pertinente sur l'ensemble des groupes. Le nombre de documents non pertinents considérés avant des documents pertinents est alors plus élevé, ce qui nuit au score d'évaluation attribué à la méthode. Par exemple, les partitionnements présentés par la figure 8.1, où les 1 représentent des documents pertinents et les 0 des documents non pertinents, obtiennent avec cette stratégie de parcours le même score qu'avec un parcours en profondeur (le score de précision moyenne est de 1 pour le clustering 1 et de 0.662 pour le clustering 2), ce qui est alors largement en faveur de la méthode 1 qui regroupe tous les documents pertinents dans un même cluster. Enfin, ce parcours est lui aussi fortement dépendant de l'ordre dans lequel sont présentés les clusters, ce qui peut grandement biaiser les évaluations réalisées.

Au vu de ces différentes observations, nous proposons la mise en place d'un certain nombre de mesures qui visent à adopter un point de vue plus global lors de l'évaluation des systèmes proposant une catégorisation des résultats, cherchant à évaluer l'ensemble de groupes plutôt qu'un groupe en particulier, sans favoriser un type de système par rapport à un autre. L'objectif est alors d'estimer les capacités réelles qu'aurait un utilisateur à atteindre les informations pertinentes à partir de la liste de clusters qui lui est présentée. Cette section se termine par une étude des approches d'évaluation proposées.

8.2.1 Parcours optimal

Face à ces limites identifiées, nous avons dans un premier temps envisagé l'introduction d'une notion de "parcours optimal", qui correspond à la meilleure liste (*i.e.,*

conduisant au meilleur score de précision moyenne) qu'il est possible de construire à partir de la liste de clusters produite par le système à évaluer. Cette liste représente le parcours effectué par un utilisateur très "chanceux", qui réalise les meilleurs changements de cluster au meilleur moment sans avoir aucune information sur les documents qu'il reste à examiner dans les différents groupes. Bien sûr, puisque très improbable, ce parcours n'est pas réaliste mais, dans un contexte d'évaluation, il peut permettre de comparer différents partitionnements de manière relativement équitable, en se basant sur le potentiel d'efficacité de chacun.

Contrairement aux approches de reconstruction de listes existantes qui requièrent la détermination d'un ordre entre les clusters, la seule contrainte imposée à ce parcours concerne l'ordre des documents à l'intérieur de chaque cluster. La liste optimale correspond alors au meilleur *shuffle* de listes C_i (ou inter-classement) qu'il est possible de produire en considérant un critère de précision moyenne (formule 3.3).

Algorithme 8.1 : Algorithme d'évaluation par construction de la liste optimale

Données : Un ensemble ordonné $\mathcal{C}$ de k listes C_i contenant chacune $|C_i|$ documents C_i^j (avec un total de n documents).
Résultat : Le score O de précision moyenne du parcours optimal.

```
1  début
2      O = 0 ; L = ( ) ; L' = ( ) ; L'' = ( ) ; stable = 0 ; i = 1 ; j = 1 ; step = 1 ; pos = 0 ;
       cur = 0;
3      Concaténation des k listes C_i : L = C_1.C_2 ... C_{k-1}.C_k;
4      O = Ap(L) ;          /* Formule 3.3 */
5      tant que stable = 0 faire
6          stable = 1;
7          pour i de 1 à k faire
8              pour j de 1 à n_i faire
9                  L'' = L;
10                 pos = Position de C_i^j dans L;
11                 pour step de 1 à (n − pos) faire
12                     pour cur de 1 à n faire
13                         L'_{Move(L,cur,pos,step)} = L_cur;          /* Formule 8.1 */
14                     fin
15                     si Ap(L') > O alors
16                         O = Ap(L') ; L'' = L' ; stable = 0;
17                     fin
18                 fin
19                 L = L'';
20             fin
21         fin
22     fin
23 fin
```

Cette liste optimale ne peut pas être obtenue en énumérant tous les *shuffles* de listes possibles (il en existe $n!/\prod_{i=1}^{k}|C_i|!$). Néanmoins, elle peut être construite relativement efficacement en concaténant les k listes C_i et en y réalisant des permutations de fragments de listes permettant une augmentation du score de précision moyenne tout en respectant la contrainte d'ordre entre les documents d'un même groupe (la liste finale L ne peut contenir un document C_i^{j+1} d'indice inférieur à un document C_i^j). L'algorithme 8.1 décrit un processus d'évaluation par construction de la liste optimale. Ce processus utilise une fonction $Move(L, cur, pos, step)$ qui retourne la nouvelle position d'un élément L_{cur} dans la liste L après avoir essayé de permuter l'élément L_{pos} avec les $step$ éléments qui le suivent dans la liste L :

$$\begin{aligned} &Move(L, cur, pos, step) = \\ &\quad \text{Si } cur \notin [pos, pos + step] \text{ alors } cur; \\ &\quad \text{Sinon } cur + Cl(L_{pos}, L_{cur}) \times (\textstyle\sum_{x=cur}^{pos+step} 1 - Cl(L_{pos}, L_x)) \\ &\qquad\qquad -(1 - Cl(L_{pos}, L_{cur})) \times (\textstyle\sum_{x=pos}^{cur} Cl(L_{pos}, L_x)). \end{aligned} \tag{8.1}$$

avec $Cl : \mathcal{D} \times \mathcal{D} \rightarrow \{0, 1\}$ une fonction qui retourne 1 si les deux documents concernés appartiennent au même cluster. Si L_{cur} n'est pas situé entre les positions pos et $pos + step$, sa position reste inchangée. Sinon, si il appartient au même groupe que L_{pos}, il est déplacé pour être positionné après tous les éléments n'appartenant pas à leur cluster et étant situés entre sa position cur et la position $pos + step$. Enfin, si L_{cur} n'appartient pas au même groupe que L_{pos}, il est avancé dans la liste pour se positionner devant tous les éléments appartenant au groupe de L_{pos} et étant situés entre les positions pos et cur. Un tel processus converge vers la liste optimale en un temps raisonnable : dans l'ensemble de nos expérimentations (voir section 8.3), l'algorithme n'a jamais requis plus de 3 itérations pour atteindre une liste stable.

Cette évaluation par construction de la liste optimale, qui ne requiert pas la détermination d'un ordre entre les différents clusters mais considère le meilleur parcours qu'il est possible d'effectuer à travers les clusters d'un partitionnement proposé, isole les performances du système considéré des biais pouvant résulter du choix d'une stratégie de parcours particulière. Elle permet alors une meilleure interprétation des résultats. Néanmoins, cette évaluation est dépendante du nombre de clusters produits par le système considéré. Par exemple, une méthode de clustering produisant autant de clusters que de documents à catégoriser ($k = n$), obtiendrait un score égal à 1. Plus le nombre de clusters produits est important, plus le score d'évaluation risque d'être élevé. Par conséquent, cette mesure ne peut être utilisée que pour comparer des systèmes présentant un même nombre de clusters à l'utilisateur. Par ailleurs, les caractéristiques du partitionnement évalué ont un impact assez important sur l'espérance du score attribué par la mesure : une méthode de clustering produisant des clusters de tailles homogènes a plus de chances d'obtenir un bon score d'évaluation qu'une méthode ayant tendance à regrouper la majorité des documents dans un même groupe. Enfin, un système ayant obtenu un score inférieur à un autre selon cette évaluation par construction de la liste optimale, peut très bien

permettre un plus grand nombre de parcours efficaces et être alors au moins aussi intéressant pour l'utilisateur. Cette évaluation n'est donc pas exempte de biais.

8.2.2 Parcours moyen

Alternativement, nous proposons de réaliser l'évaluation d'un partitionnement par considération du parcours "moyen" qu'il est possible d'effectuer à travers les groupes proposés. Cela revient à considérer l'espérance mathématique du score de précision moyenne pour un parcours effectué par un utilisateur "aveugle" (*i.e.,* qui n'oriente pas sa recherche selon les informations qu'il a déjà collectées dans les différents groupes). Étant donnée une fonction $exam : \{0, \ldots, n-1\} \times \{0, \ldots, |\mathcal{P}ert|\} \rightarrow \mathcal{D}$, définie telle que $exam(t, p)$ retourne le prochain document examiné après avoir déjà rencontré t documents dont p sont pertinents [3], cette espérance peut être calculée de la manière suivante [4] :

$$ExpAP(\mathcal{C}) = \frac{\sum_{t=0}^{n-1} \sum_{i=1}^{k} \sum_{j=1}^{|C_i|} \sum_{p=0}^{|\mathcal{P}ert|} Pert(C_i^j) \times P(exam(t,p) = C_i^j) \times \frac{p+1}{t+1}}{|\mathcal{P}ert|} \tag{8.2}$$

Toute la difficulté réside alors dans le calcul de la probabilité $P(exam(t, p) = C_i^j)$. Il n'est en effet pas raisonnable de chercher à l'obtenir en testant, pour tous les documents et toutes les positions t et p envisageables, toutes les possibilités de parcours partiels permettant d'examiner C_i^j après t documents dont p pertinents.

Afin de réduire le nombre de calculs à effectuer, nous choisissons alors de travailler avec des configurations $\vec{x}$, dont les composantes x_i (avec $i \in \{1, \ldots, k\}$) représentent les nombres de documents déjà examinés dans chaque cluster C_i (et respectent donc l'axiome suivant : $\forall i \in \{1, \ldots, k\}, x_i \leq |C_i|$). Avec $\mathcal{X}$ l'ensemble de toutes les configurations possibles, nous considérons alors une fonction $sel : \mathcal{X} \rightarrow \mathcal{C}$, de sélection du prochain cluster à parcourir à partir d'une configuration donnée, qui nous permet de définir la probabilité de sélectionner un cluster C_i (avec $i \in \{1, \ldots, k\}$) selon une configuration $\vec{x}$ de documents déjà examinés [5] :

$$P(sel(\vec{x}) = C_i) = \frac{1}{|\{j \in \{1, \ldots, k\}, x_j < |C_j|\}|} \tag{8.3}$$

De manière à évaluer la probabilité de rencontrer une configuration $\vec{x}$ donnée, nous définissons alors une notion de voisinage entre configurations par le biais d'une fonction $V : \mathcal{X} \times \mathcal{C} \rightarrow \mathcal{X}$ telle que $V(\vec{x}, C_i)$ représente la configuration permettant

3. $\mathcal{P}ert \subseteq \mathcal{D}$ représente le sous-ensemble des documents pertinents contenus dans l'ensemble $\mathcal{D}$ des n documents partitionnés. La fonction $Pert : \mathcal{D} \rightarrow \{0, 1\}$, qui lui est associée, retourne 1 si le document considéré appartient à ce sous-ensemble (et 0 sinon).

4. $P(A)$ dénote, de manière classique, la probabilité de l'évènement A.

5. Où $|\{j \in \{1, \ldots, k\}, x_j < |C_j|\}|$ correspond au nombre de groupes n'ayant pas encore été entièrement examinés dans la configuration $\vec{x}$.

d'atteindre $\vec{x}$ en examinant le premier document non encore examiné dans le cluster C_i :

$$V(\vec{x}, C_i) = (y_1, \ldots, y_k) \mid y_i = x_i - 1 \ \wedge \ \forall j \in \{1, \ldots, k\}, j \neq i \Rightarrow y_j = x_j \qquad (8.4)$$

Ainsi, la probabilité de rencontrer une configuration donnée $\vec{x}$ peut être évaluée par [6] :

$$P(\vec{x}) = \sum_{i \in \{1,\ldots,k\}, x_i > 0} P(V(\vec{x}, C_i)) \times P(sel(V(\vec{x}, C_i)) = C_i) \qquad (8.5)$$

Il est alors possible de calculer la probabilité $P(exam(t,p) = C_i^j)$ qui nous intéresse :

$$P(exam(t,p) = C_i^j) = \sum_{\vec{x} \in config(C_i^j, t, p)} P(\vec{x}) \times P(sel(\vec{x}) = C_i) \qquad (8.6)$$

où $config(C_i^j, t, p)$ correspond à l'ensemble des configurations permettant d'examiner C_i^j après avoir déjà rencontré t documents dont p pertinents, qui est définie par une fonction $config : \mathcal{D} \times \{0, \ldots, n-1\} \times \{0, \ldots, |\mathcal{P}ert|\} \rightarrow 2^{\mathcal{X}}$ telle que :

$$\begin{aligned} config(C_i^j, t, p) = \ & \{(x_1, \ldots, x_k) \in \mathcal{X} \mid x_i = (j-1) \wedge \\ & \textstyle\sum_{l=1}^{k} x_l = t \ \wedge \ \sum_{a=1}^{k} \sum_{b=1}^{x_a} Pert(C_a^b) = p\} \end{aligned} \qquad (8.7)$$

De tels calculs restent relativement complexes, mais leur emploi pour l'évaluation de partitionnements de 50 documents paraît tout à fait envisageable (autour de 10 secondes sur un *Pentium 4, 3GHz PC*). Par ailleurs, pour les partitionnements de plus larges ensembles de documents, le score $ExpAP$ peut être estimé statistiquement, en effectuant des parcours aléatoires à travers les clusters [7] et en calculant la moyenne des précisions moyennes obtenues sur les listes résultant de ces différents parcours. Des expérimentations [8] ont montré que cette estimation se révélait très proche du score réel que l'on aurait pu obtenir avec la formule 8.2 et très robuste pour des nombres de documents considérés supérieurs à 50, quels que soient le corpus utilisé et le nombre de clusters produits : le plus grand écart enregistré entre deux estimations d'une même instance (100 estimations, comprenant chacune 1000 parcours aléatoires, ont été réalisées pour chaque instance) est égal à 0.0012 pour des clusterings aléatoires de 100 documents et de 0.0008 pour des clusterings aléatoires de 500. Étant donnés les écarts observés entre scores obtenus sur des instances différentes, ces écarts paraissent tout à fait négligeables. Cette estimation statistique peut alors

6. Il est à noter que $P((0, \ldots, 0)) = 1$.

7. Afin de respecter les différentes probabilités, le processus simule le parcours d'un utilisateur "aveugle" : plutôt que de choisir une route donnée parmi l'ensemble des parcours possibles, le processus construit le parcours itérativement, en sélectionnant aléatoirement, à chaque itération, un cluster dans lequel examiner le premier document non encore rencontré.

8. Les corpus utilisés dans ces expérimentations sont ceux décrits en section 3.2.

être utilisée lorsque les calculs de l'espérance de précision moyenne d'un parcours de clusters paraît trop complexe, pour des nombres de documents considérés supérieurs à 50.

Cette évaluation par calcul de l'espérance du score de précision moyenne pour un partitionnement donné est indépendante du nombre de clusters produits par le système considéré. En effet, si le nombre de routes efficaces augmente avec le nombre de clusters produits, le nombre de mauvaises routes augmente aussi, ce qui permet de trouver un équilibre qui autorise la comparaison de scores obtenus sur des partitionnements aux nombres de clusters différents. Avec une telle évaluation, seules comptent les positions des documents pertinents dans les listes correspondant aux clusters considérés. Le parcours moyen, qui résume l'ensemble des parcours possibles, représente un bon compromis entre parcours en profondeur et parcours en largeur. Par ailleurs, une telle mesure ne souffre pas des biais induits par la détermination d'un ordre entre les clusters et permet alors certainement de réaliser une évaluation plus fiable que les deux approches de reconstruction en largeur et en profondeur. D'un point de vue plus pragmatique, cette mesure rend compte de la capacité qu'aura un utilisateur novice (qui n'a pas l'habitude d'utiliser un système présentant des catégories de résultats et donc ne s'oriente pas selon les informations collectées dans les différents groupes) à atteindre les documents pertinents à partir de la liste de clusters présentés.

8.2.3 Parcours orienté par la pertinence des documents

Les jugements de pertinence, utilisés pour orienter la stratégie de parcours proposée dans [Leuski, 2001a], peuvent être inclus dans le calcul du score d'évaluation du parcours moyen afin de simuler un parcours réalisé par un utilisateur plus expérimenté. Dans ce sens, afin d'orienter plus probablement la recherche vers des clusters qui, selon leurs documents déjà examinés, présentent un fort ratio de documents pertinents, la probabilité $P(sel(\vec{x}) = C_i)$, de choisir le cluster Ci après avoir examiné x_l documents dans chaque cluster C_l, peut être remplacée dans les formules 8.5 et 8.6 par :

$$P(sel(\vec{x}) = C_i) = \frac{0.5 + \sum_{j=1}^{x_i} Pert(C_i^j)}{x_i + 1} \Big/ \sum_{l \in \{1,\ldots,k\}, x_l < |C_l|} \frac{0.5 + \sum_{j=1}^{x_l} Pert(C_l^j)}{x_l + 1} \quad (8.8)$$

Une telle probabilité de sélection de cluster permet de déterminer un score d'évaluation correspondant à l'espérance de précision moyenne obtenue par un utilisateur se servant de ses observations pour s'orienter vers les clusters semblent contenir le plus fort ratio de documents pertinents. Au début du processus, tous les clusters possèdent la même probabilité de sélection. Selon le nombre et les positions des documents pertinents qu'ils contiennent, la probabilité de sélection de chaque cluster évolue au fur et à mesure que de nouveaux documents sont examinés. Contrairement

à la stratégie de recherche proposée dans [Leuski, 2001a], ce parcours moyen orienté par des retours de pertinence ne requiert pas, encore une fois, de déterminer un ordre entre les clusters. Les ratios de documents pertinents dans chaque cluster sont considérés en parallèle, ce qui permet d'écarter le biais, énoncé plus haut pour la stratégie de recherche proposée dans [Leuski, 2001a], concernant l'intervalle existant entre le dernier examen d'un document pertinent dans un cluster donné et la prise de décision de changer de cluster.

8.2.4 Parcours orienté par la proximité des documents pertinents

Tel que mentionné précédemment, un moyen souvent employé pour présenter les clusters à un utilisateur est de sélectionner un document représentant dans chacun d'entre eux. Dans ce cas, les documents sont généralement ordonnés à l'intérieur de chaque cluster selon leur similarité avec le représentant de leur groupe. Il paraît en effet naturel de commencer la présentation des documents d'un cluster par ceux qui ressemblent le plus au représentant qui a attiré l'attention de l'utilisateur. Avec une telle présentation des résultats, plus les représentants des clusters sont proches des documents pertinents, plus ils ont de chances d'être intéressants pour l'utilisateur, et plus les documents pertinents sont alors susceptibles d'être atteints facilement. Cette observation nous a conduit à considérer un nouveau critère d'évaluation, la proximité des pertinents PrP, pour estimer la capacité qu'aura un utilisateur à atteindre les informations pertinentes à partir de la liste de représentants qui lui est présentée. Ce critère correspond à la proximité moyenne (selon la mesure de similarité utilisée pour ordonner les documents [9]) des documents pertinents avec les représentants de leur cluster :

$$PrP = \frac{\sum_{i=1}^{n} Pert(D_i) \times Sim(D_i, Rep(D_i))}{\sum_{i=1}^{n} Pert(D_i)} \tag{8.9}$$

avec $Rep : \mathcal{D} \rightarrow \{C_1^1, \ldots, C_k^1\}$ une fonction retournant le représentant du cluster contenant le document considéré et $Sim(D_x, D_y)$ la similarité du document D_x avec le document D_y. Pour les documents à la fois pertinents et représentants de groupe, $D_i = Rep(D_i)$ et donc $Sim(D_i, Rep(D_i)) = Sim(D_i, D_i)$, ce qui revient à considérer la similarité d'un document avec lui même. Le fait d'utiliser la valeur 1 pour une telle similarité conduit à donner un poids trop important aux documents pertinents étant représentants de groupe et donc le score d'évaluation obtenu risque de dépendre quasiment uniquement du nombre de documents pertinents à avoir été sélectionnés en tant que représentant de leur cluster. Par conséquent, nous préférons fixer cette similarité (entre un document et lui même) de telle sorte qu'elle corresponde à la similarité maximale que le document concerné possède avec un autre document de son groupe. Cela permet de donner un poids important aux représentants pertinents

9. Dans nos expérimentations, nous utilisons la mesure de *Cosine* normalisée par régression statistique que nous avons proposée en section 7.1

(ce qui paraît naturel puisque ces documents sont les premiers à être examinés par l'utilisateur) tout en permettant aux représentants non pertinents d'avoir un impact positif sur le score d'évaluation s'ils sont fortement corrélés aux documents pertinents de leur groupe. Puisque le score dépend des connections entre représentants et pertinents, cette mesure permet d'évaluer le contenu des documents initialement présentés à l'utilisateur. Même si certains représentants n'ont pas été jugés pertinents lors de l'annotation du corpus, ils peuvent aborder un aspect intéressant du sujet et permettre à un utilisateur d'identifier leur groupe comme probablement pertinent. Ce qui est évalué ici correspond alors au degré de facilité avec laquelle un utilisateur retrouvera les informations qu'il recherche à partir des informations contenues par les représentants qui lui sont présentés. Il est à noter qu'un système regroupant tous les documents pertinents dans un même groupe n'est pas nécessairement pénalisé par cette mesure par rapport à un système qui distribuerait les documents pertinents sur l'ensemble des groupes puisque ce dernier ne bénéficie pas des similarités potentiellement élevées existant entre les documents pertinents qu'il affecte à des groupes différents. Tout dépend du degré de relations entre les documents pertinents : s'ils traitent tous d'un même aspect du sujet, la configuration produite par la méthode 1 de la figure 8.1 sera certainement préférée. Si au contraire, ils abordent des thématiques bien différentes (il paraît alors naturel de préférer un système qui puisse les distinguer), notre critère d'évaluation aura tendance à favoriser un système proposant un partitionnement tel que celui présenté par la méthode 2 de la même figure. Notons néanmoins que ce critère dépend du nombre de clusters produits par le système.

Tel que c'est le cas pour la prise en compte des retours de pertinence, il est possible d'inclure une considération du contenu des documents examinés dans le calcul du score d'évaluation du parcours moyen. Il s'agit alors d'orienter plus probablement la recherche vers des clusters dont le contenu des documents examinés semble correspondre aux informations portées par les documents pertinents. La probabilité $P(sel(\vec{x}) = C_i)$, de choisir le cluster C_i après avoir examiné x_l documents dans chaque cluster C_l, peut alors être remplacée, dans les formules 8.5 et 8.6, par :

$$P(sel(\vec{x}) = C_i) = \frac{\dfrac{0.5 + \sum_{j=1}^{x_i} \sum_{D \in \mathcal{P}ne(\vec{x})} \dfrac{Sim(D, C_i^j)}{|\mathcal{P}ne(\vec{x})|}}{x_i + 1}}{\sum_{l \in \{1,\ldots,k\}, x_l < |C_l|} \dfrac{0.5 + \sum_{j=1}^{x_l} \sum_{D \in \mathcal{P}ne(\vec{x})} \dfrac{Sim(D, C_l^j)}{|\mathcal{P}ne(\vec{x})|}}{x_l + 1}} \tag{8.10}$$

où $\mathcal{P}ne(\vec{x})$ correspond à l'ensemble des documents pertinents n'ayant pas encore été examinés dans la configuration $\vec{x}$. Les probabilités de sélection des clusters dépendent alors des similarités moyennes des documents examinés dans chacun d'entre eux avec les documents pertinents n'ayant pas encore été rencontrés. Afin de travailler avec

des valeurs de similarité suffisamment dispersées pour que les différences puissent influer sur la recherche, le processus d'évaluation commence par identifier les similarités minimales et maximales de chaque document pertinent et utilise ces valeurs pour répartir ses similarités sur $[0, 1]$. Avec une telle normalisation, le document D_x le plus proche du document pertinent D_p se voit attribuer une similarité $Sim(D_x, D_p) = 1$ et le document D_y qui en est le plus éloigné obtient une similarité $Sim(D_y, D_p) = 0$, ce qui permet de donner un poids plus important aux clusters contenant des documents très proches des documents pertinents. De la même façon qu'avec le parcours moyen utilisant des jugements de pertinence, tous les clusters possèdent initialement la même probabilité d'être sélectionnés. Ces probabilités évoluent ensuite selon le contenu des documents rencontrés dans chaque cluster. Le fait de ne considérer que les documents pertinents non rencontrés permet d'orienter la recherche vers d'autres clusters lorsque tous les documents pertinents connectés à la thématique d'un cluster donné ont déjà été examinés. Ce critère d'évaluation, fondé sur la proximité des pertinents plutôt que sur des retours binaires de pertinence, prend alors en compte les informations contenues par l'ensemble des documents, qu'ils soient pertinents ou non, pour orienter la recherche vers les clusters les plus pertinents, ce qui nous semble mieux correspondre aux comportements réels des utilisateurs.

Certains systèmes présentent les clusters de documents par des labels décrivant leur contenu (voir section 2.4). L'utilisateur peut alors s'appuyer sur ces descriptions pour choisir les clusters qui lui semblent le mieux correspondre à ses besoins. Pour effectuer une évaluation réaliste, il est envisageable, dans le cas de tels systèmes, d'inclure une prise en compte de ces labels dans les calculs des probabilités de sélection donnés par la formule 8.10. Une initialisation du numérateur et du dénominateur de cette formule selon la similatité des documents pertinents avec les descriptions des clusters (plutôt que selon la valeur arbitraire de 0.5) peut alors permettre de favoriser les clusters paraissant les plus interessants :

$$P(sel(\vec{x}) = C_i) = \frac{\dfrac{\displaystyle\sum_{D \in \mathcal{P}ne(\vec{x})} \frac{Sim(D, Label(C_i))}{|\mathcal{P}ne(\vec{x})|} + \sum_{j=1}^{x_i} \sum_{D \in \mathcal{P}ne(\vec{x})} \frac{Sim(D, C_i^j)}{|\mathcal{P}ne(\vec{x})|}}{x_i + 1}}{\displaystyle\sum_{l \in \{1,\ldots,k\}, x_l < |C_l|} \frac{\displaystyle\sum_{D \in \mathcal{P}ne(\vec{x})} \frac{Sim(D, Label(C_l))}{|\mathcal{P}ne(\vec{x})|} + \sum_{j=1}^{x_l} \sum_{D \in \mathcal{P}ne(\vec{x})} \frac{Sim(D, C_l^j)}{|\mathcal{P}ne(\vec{x})|}}{x_l + 1}} \tag{8.11}$$

Avec des systèmes décrivant le contenu des clusters en sélectionnant un document représentatif dans chacun d'entre eux (tel que réalisé pour nos expérimentations), il est probable que les utilisateurs examinent l'ensemble des représentants de cluster avant de se lancer dans le parcours des groupes qui leur sont proposés. Les k

représentants de cluster sont alors susceptibles d'être les k premiers documents à être rencontrés (probablement dans l'ordre où ils sont présentés). Pour être plus réaliste, notre mesure d'évaluation peut alors inclure cette observation dans ses calculs, en fixant $P((1,\ldots,1)) = 1$ et en considérant $P(exam(i-1, \sum_{j=1}^{i-1} Pert(C_j^1)) = C_i^1) = 1$ pour chacun des représentants de cluster [10] (les autres probabilités d'examen sont fixées à 0 pour ces documents). Néanmoins, étant donné le faible impact qu'elle a sur les résultats finaux [11], une telle orientation de la recherche peut être omise.

8.2.5 Étude des mesures proposées

Nous proposons ici de réaliser un certain nombre d'expérimentations visant à rendre compte des tendances observées par les mesures proposées face à des partitionnements de types différents. Dans cette étude, ainsi que dans la suite de ce rapport, nous utiliserons les notations données dans le tableau 8.1.

EcD	Écart-type du nombre de documents par cluster	
NbP	Nombre de représentants pertinents	
$MK1$	Qualité du groupe optimal	(section 3.3.2)
PrP	Proximité des documents pertinents	(section 8.2.4)
PRO	Parcours en profondeur	(section 3.3.2)
LAR	Parcours en largeur	(section 3.3.2)
LEU	Parcours proposé dans [Leuski, 2001a]	(section 8.2)
OPT	Parcours optimal	(section 8.2.1)
$PM1$	Parcours moyen	(section 8.2.2)
$PM2$	Parcours orienté par la pertinence des documents	(section 8.2.3)
$PM3$	Parcours orienté par la proximité des documents pertinents	(section 8.2.4)

Table 8.1 – Notations des mesures d'évaluation

Dans les expérimentations réalisées, nous avons choisi d'ordonner les documents de chaque cluster selon leur similarité avec leur représentant (en terme de *NCosine*, voir section 7.1). Chaque représentant correspond au document le plus proche de la requête parmi les documents contenus dans le cluster concerné. Lorsque nécessaire (c'est à dire pour PRO, LAR et LEU), un ordre entre les groupes de documents est déterminé selon la similarité du représentant de chaque groupe avec la requête (en terme de *NCosine*, voir section 6.2.2).

10. En supposant d'avoir placé les représentants en début de cluster : $\forall_{i\in\{1,\ldots,k\}, Rep(C_i)=C_i^1}$

11. Les premiers documents des clusters ont, dans tous les cas, une forte probabilité d'être examinés en début de recherche.

Degré de corrélation avec le comportement des utilisateurs

L'objectif des mesures est de rendre compte de la capacité qu'aura un utilisateur à atteindre les documents qu'il recherche à partir de la liste de clusters qui lui est présentée. Afin d'être de bons indicateurs de cette capacité, les parcours de clusters doivent se rapprocher au maximum des routes qu'un utilisateur réel aurait pu suivre. Nous proposons alors ici d'évaluer la corrélation entre les précisions moyennes des listes résultants des différents parcours proposés avec celles obtenues sur les listes résultant des parcours réalisés par différents individus sur des partitionnements produits par quatre méthodes de clustering appliquées aux 20 premiers résultats retournés par une recherche préliminaire (utilisant la mesure *NCosine*) :

- G, N : la méthode hiérarchique *Group-Average* utilisant la mesure *NCosine* ;
- K, N : la méthode *K-Means* utilisant la mesure *NCosine* ;
- G, Q : la méthode hiérarchique *Group-Average* utilisant la mesure *QSSM* ;
- K, Q : la méthode *K-Means* utilisant la mesure *QSSM*.

Nous avons demandé à dix personnes volontaires de chercher à localiser le plus rapidement possible des documents pertinents pour différentes requêtes en utilisant les partitionnements produits par ces quatre différentes méthodes de clustering. Dans le but d'obtenir des résultats représentatifs de recherches réelles, nous leur avons expliqué que les différents groupes présentés étaient supposés représenter différents aspects de l'ensemble des documents considérés et qu'ils devaient alors lire attentivement chacun des documents qui leur seraient présentés pour orienter leur recherche vers les groupes qui leur semblent le plus probablement contenir les informations pertinentes. Pour chaque requête considérée [12], chaque système expérimenté [13] et chaque individu, le processus expérimental est le suivant :

1. La description narrative de la requête est présentée au sujet qui la lit pour saisir précisément les informations qu'il doit rechercher ;
2. Les k (dans nos expérimentations, $k = 3$) représentants de cluster sont présentés au sujet qui les lit pour avoir une idée des différences entre chaque groupe ;
3. Le sujet choisit un index de cluster (parmi ceux des clusters non encore entièrement examinés) en fonction de son sentiment concernant le cluster étant le plus susceptible de contenir les informations qu'il recherche ;
4. Le premier document non encore examiné du cluster choisi est présenté et le sujet le lit pour saisir de quoi il retourne, ce qui est susceptible de changer ses impressions sur le contenu des différents groupes ;
5. Si il reste des documents pertinents à localiser, le processus se poursuit à l'étape 3. Sinon, la précision moyenne de la liste résultant du parcours réalisé

12. Les topics 1-20 de *TREC-1* sont utilisés dans cette étude.

13. Chaque individu est confronté successivement aux quatre systèmes *G,N*, *K,N*, *G,Q* et *K,Q* pour chacune des requêtes. Néanmoins, l'ordre dans lequel ces systèmes sont utilisés est différent pour chaque sujet. Le corpus utilisé est le corpus *AP*, qui contient les documents les plus courts (voir section 3.2).

est calculée et le processus se termine.

Même si les différents sujets ont parfois suivi des routes différentes, les expérimentations réalisées permettent de capturer les variations du degré de capacité à atteindre les documents pertinents à partir des listes de représentants présentées. L'objectif est alors de calculer la corrélation entre les variations observées et celles générées par les différentes mesures de reconstruction de liste. La corrélation entre valeurs observées o_i et valeurs mesurées m_i est évaluée par un coefficient $R^2_{o,m}$:

$$R^2_{o,m} = 1 - \frac{\sum_{i=1}^{N}(o_i - m_i)^2}{\sum_{i=1}^{N}(o_i - \bar{o})^2} \tag{8.12}$$

où o_i correspond à la moyenne des scores obtenus par les 10 sujets sur la i-ième requête en utilisant le système considéré (m_i le score obtenu par la mesure concernée sur cette même requête) et $\bar{o}$ représente la moyenne des scores o_i sur l'ensemble des 20 requêtes. Ce que l'on cherche à évaluer ici est la capacité des mesures proposées à suivre les variations des scores obtenus par les sujets. Puisque le but final est de déterminer si les mesures sont capables d'ordonner correctement les différents systèmes selon leurs performances, le fait qu'une mesure obtienne des scores plus élevés qu'une autre ne nous importe pas, seule compte la corrélation avec les variations de score observées. Par conséquent, plutôt que de considérer directement les valeurs obtenues, nous normalisons chaque distribution de scores en remplaçant chaque valeur s_i (observée ou obtenue par une mesure) par :

$$s_i = \frac{s_i - \bar{s}}{\sigma_s} \tag{8.13}$$

où $\bar{s}$ correspond à la moyenne des scores sur l'ensemble des requêtes et σ_s à leur écart-type. En utilisant une telle normalisation, la corrélation évaluée ne dépend que des variations observées (puisque chaque distribution présente alors une moyenne de 0 et un écart-type de 1). Le tableau 8.2 présente, pour les quatre différents systèmes, les coefficients de corrélation obtenus entre les distributions de valeurs observées et celles des valeurs calculées par les différentes approches de reconstruction de liste.

$R^2_{o,m}$	PRO	LAR	LEU	OPT	PM1	PM2	PM3
G, N	0.85	0.80	0.89	0.87	0.83	0.92	0.92
K, N	0.51	0.76	0.62	0.73	0.81	0.86	0.90
G, Q	0.72	0.72	0.83	0.82	0.78	0.89	0.91
K, Q	0.47	0.71	0.55	0.70	0.74	0.81	0.88

TABLE 8.2 – Corrélation entre variations observées

Ainsi que nous le verrons en section 8.3, la méthode *Group-Average* conduit bien souvent à l'obtention d'un cluster contenant un très grand nombre de documents. Les choix proposés aux utilisateurs sont alors bien moins nombreux puisque les autres clusters sont souvent très vite épuisés et qu'il ne reste alors plus qu'un seul cluster contenant des documents non examinés. Il est alors bien plus aisé de suivre les variations observées. Dans ce cas, le fait que le parcours en profondeur obtienne un meilleur coefficient de corrélation que le parcours en largeur s'explique par le fait que les documents contenus par les plus petits clusters sont souvent bien marginaux et n'ont alors pas grand chose à voir avec le sujet de la recherche. Avec l'algorithme *K-Means* qui produit des clusters de tailles plus homogènes, il est plus difficile de suivre une route optimale puisque les documents pertinents sont mieux répartis dans les clusters. Cette difficulté est par ailleurs accrue lors de l'utilisation de la mesure $QSSM$[14]. Les coefficients de corrélation reportés montrent que notre parcours orienté par la pertinence des documents $PM2$, et *a fortiori* orienté par la proximité des documents pertinents $PM3$, parait bien mieux suivre les variations de score observées avec les individus impliqués dans l'étude que les autres approches d'évaluation quel que soit le système utilisé. Ils sont donc susceptibles d'être de meilleurs indicateurs de la capacité à atteindre les documents pertinents à partir d'une liste de clusters présentée à l'utilisateur.

Influence du degré de répartition des documents

Afin d'évaluer l'influence qu'ont les différentes caractéristiques des clusterings considérés (*i.e.,* les variations de tailles de leurs clusters et le degré de répartition des documents pertinents dans leurs clusters) sur les espérances des scores obtenus par les différentes mesures, nous proposons de réaliser des expérimentations impliquant des clusterings aléatoires de documents. Différents types de clustering sont produits pour étudier l'influence du degré de répartition des documents dans les clusters : chaque type de clustering affecte chacun des documents dans un cluster spécifique (le premier par exemple) selon une probabilité donnée, prise parmi les valeurs $\{0.2, 0.33, 0.5, 0.66, 1\}$. Alors qu'une probabilité de 1 conduit au regroupement de l'ensemble des documents dans un même groupe, une probabilité de 0.2 tend à produire des groupes de tailles relativement homogènes. Pour chaque requête (topics 1-50 de *TREC*), 100 clusterings de chaque type ont alors été produits pour étudier l'influence des tailles de clusters sur les scores attribués par les mesures d'évaluation. Ces clusterings concernent la répartition en cinq clusters des 20 premiers documents retournés par une recherche initiale sur le corpus AP. Les représentants de clusters sont ici sélectionnés aléatoirement. La figure 8.2 trace les distributions des scores obtenus par les différentes mesures pour les différents types de clustering.

14. Il est à noter que cette difficulté plus élevée ne traduit en rien une moins bonne efficacité du système, il ne s'agit ici que de la difficulté à trouver la meilleure route possible, et non de la difficulté accrue à atteindre les documents pertinents.

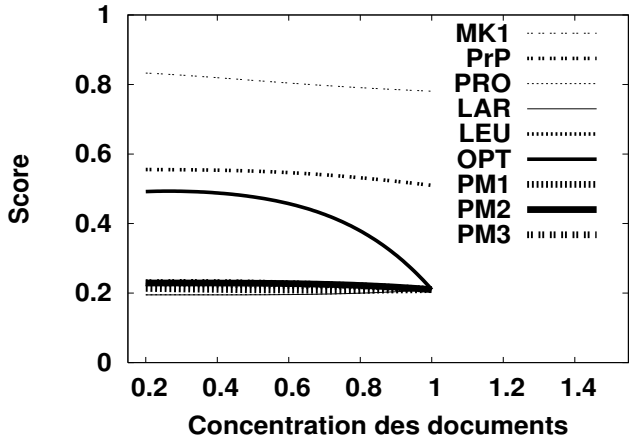

FIGURE 8.2 – Influence de la répartition des documents dans les clusters

Selon les courbes présentées, à l'exception du parcours optimal OPT et de la proximité des pertinents PrP dont les espérances diminuent pour les clusterings regroupant la plupart des documents dans un même groupe, l'ensemble des mesures paraissent présenter une espérance constante quelle que soit la diversité de tailles de clusters. Plus intéressante est l'étude de l'influence du degré de répartition des documents pertinents dans les clusters : après avoir affecté l'ensemble des documents non pertinents aux différents clusters, chaque document pertinent est alloué au premier cluster avec une probabilité de 0.2, 0.33, 0.5, 0.66 ou 1 selon le type de partitionnement concerné. Ces différentes probabilités d'affectation des documents pertinents au premier groupe nous permettent d'obtenir des clusterings présentant différents degrés de concentration de l'information pertinente : alors qu'une probabilité de 1 conduit à la production d'un cluster contenant la totalité des documents pertinents, une probabilité de 0.2 favorise la distribution de l'information pertinente. La figure 8.3 présente alors les distributions des scores obtenus par les mesures sur ces différents types de clustering.

Tel qu'attendu, l'espérance du parcours en profondeur augmente avec la probabilité d'affecter tous les documents pertinents dans un même cluster. L'espérance du parcours en largeur diminue dans le même temps. Conformément à notre intuition, le parcours proposé par [Leuski, 2001a] semble largement favoriser les clusterings rassemblant tous les documents pertinents dans le même groupe. Selon les courbes présentées par la figure 8.3, nos parcours $PM2$ et $PM3$ paraissent constituer les mesures les plus équitables puisque leur espérance ne semble que très légèrement affectée par les variations du degré de répartition des documents pertinents dans les clusters. L'espérance du parcours $PM3$ semble néanmoins diminuer légèrement lorsque le degré de concentration de l'information pertinente augmente. Cela peut s'expliquer par le fait que les clusters considérés ne représentent pas de réelles thématiques

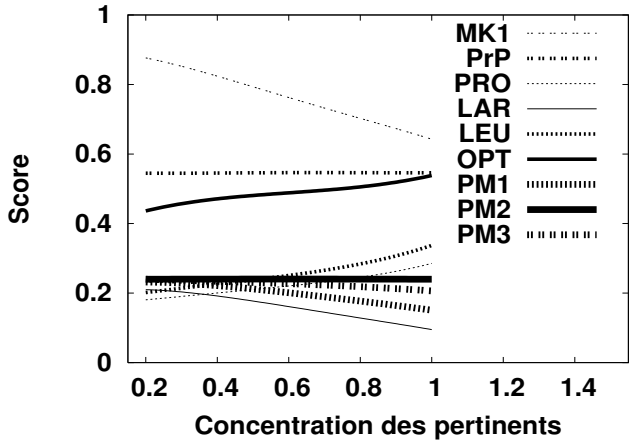

FIGURE 8.3 – Influence de la répartition des documents pertinents dans les clusters

puisqu'ils ont été produits de manière aléatoire. Ce parcours a alors des difficultés à déterminer les clusters les plus potentiellement intéressants à partir des similarités des représentants avec les documents pertinents. Nous pensons que cette diminution d'espérance n'existerait pas avec des partitionnements produits par des méthodes de clustering considérant les similarités existant entre les documents. Selon les résultats obtenus, on peut légitimement considérer les parcours $PM2$ et $PM3$ comme les plus à même de comparer des systèmes produisant des partitionnements de types différents.

Influence du nombre de clusters produits

Afin d'évaluer l'influence du nombre de clusters produits sur l'espérance des différentes mesures, nous produisons, pour chaque requête (topics 1-50 de *TREC*) et chaque nombre de clusters entre 3 et 7, 100 clusterings aléatoires des 50 premiers documents retournés par une recherche initiale. Ces clusterings sont produits en choisissant aléatoirement k documents (avec k le nombre de clusters à produire), qui représentent alors les centres des clusters, et en affectant chaque document non sélectionné au cluster dont le centre lui est le plus proche. Un tel processus nous permet de travailler avec des clusters n'étant pas totalement dénués de signification, tout en écartant les biais qui auraient pu survenir en choisissant de travailler avec une méthode de clustering particulière (qui peut favoriser un nombre de clusters donné selon les distributions de similarités considérées). La figure 8.4 trace les distributions de scores obtenus par les différentes mesures selon le nombre de clusters produits. Selon les courbes obtenues, on peut noter que la plupart des mesures ne sont pas influencées par le nombre de clusters produits (à l'exception de LAR, OPT, PrP et $PM1$). La diminution d'espérance du parcours moyen lorsque le nombre de clusters augmente peut s'expliquer par le fait que plus le nombre de

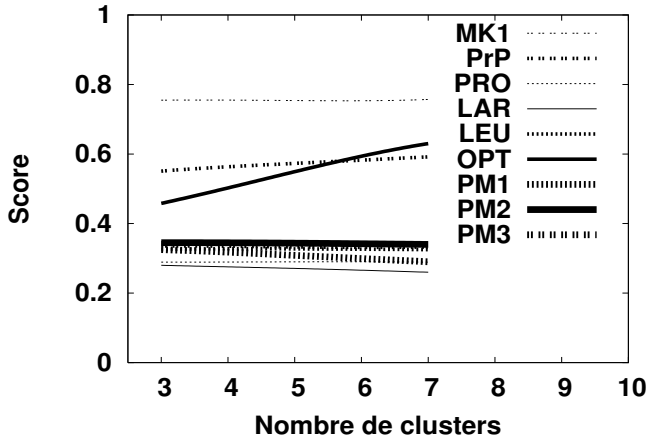

FIGURE 8.4 – Influence du nombre de clusters produits

clusters est élevé, plus le nombre de clusters ne contenant pas de document pertinent est susceptible d'être élevé, d'autant plus lorsque l'on considère la *Cluster Hypothesis* qui fait tendre la majorité des documents pertinents à se regrouper dans un même groupe. L'utilisation des retours de pertinence $PM2$ ou de la proximité des documents examinés avec les documents pertinents $PM3$ pour orienter le parcours semble résoudre ce problème. Les informations collectées par ces mesures au cours de la recherche permettent d'orienter leur parcours vers les clusters contenant les documents pertinents. Observant une espérance relativement constante quel que soit le nombre de clusters produits, ces mesures peuvent être utilisées pour comparer des systèmes réalisant des clusterings de résultats aux nombres de groupes différents.

8.3 Comparaison des systèmes

L'objectif de cette section est de comparer différents systèmes, en utilisant les nouvelles mesures proposées dans la précédente section, afin de mesurer l'impact que peut avoir la prise en compte de la requête sur les clusters présentés à l'utilisateur. Les différences entre les systèmes comparés résident dans la méthode de clustering employée (*Group-Average* ou *K-Means*) et dans la mesure de similarité utilisée (*NCosine* ou *QSSM*). Les résultats expérimentaux donnés dans cette section concernent tous des partitionnements, en cinq clusters, des 50 premiers résultats d'une recherche initiale (utilisant la mesure *NCosine*). Les expérimentations sont réalisées sur les corpus présentés en section 7.2.2 avec les requêtes 1 à 50 de *TREC* (voir section 3.2).

Le tableau 8.3 présente les statistiques des partitionnements produits par les différents systèmes : Co correspond à la cohésion moyenne des groupes formés (formule 2.2), Sep à leur séparation (c'est à dire la dissimilarité moyenne entre groupes,

		ZIFF			AP			WSJ			FR		
		Co	*Sep*	*EcD*	*Co*	*Sep*	*EcD*	*Co*	*Sep*	*EcD*	*Co*	*Sep*	*EcD*
Tit	G, N	0.56	0.47	15.94	0.57	0.47	15.32	0.57	0.47	16.10	0.65	0.44	13.04
	K, N	0.57	0.45	7.06	0.58	0.45	6.50	0.58	0.45	6.00	0.66	0.41	6.28
	G, Q	0.56	0.46	13.62	0.57	0.46	14.87	0.57	0.46	14.51	0.65	0.43	13.17
	K, Q	0.56	0.45	7.03	0.57	0.45	7.17	0.58	0.45	6.37	0.64	0.42	6.93
Nar	G, N	0.58	0.46	14.01	0.59	0.46	12.84	0.58	0.46	13.20	0.61	0.46	14.28
	K, N	0.58	0.45	5.82	0.59	0.45	5.86	0.59	0.45	5.87	0.63	0.44	6.02
	G, Q	0.58	0.45	13.10	0.59	0.45	12.50	0.58	0.45	12.43	0.61	0.45	12.63
	K, Q	0.58	0.45	5.86	0.59	0.45	5.96	0.58	0.45	5.53	0.63	0.44	6.10

TABLE 8.3 – Statistiques des partitionnements produits

voir formule 2.3) et *EcD* à l'écart type entre tailles de clusters. Les critères de cohésion et de séparation des clusters sont calculés selon la mesure *NCosine*. Selon les résultats, la méthode *Group-Average* tend à produire un cluster contenant une grande part des documents et à isoler les documents marginaux dans les autres groupes, ce qui conduit à l'obtention de groupes très dissimilaires mais présentant un faible degré de cohésion. La méthode *K-means*, quant à elle, paraît produire des clusters de tailles plus homogènes, dont les éléments sont plus proches les uns des autres. Enfin, la mesure *QSSM* semble conduire à une légère baisse des deux critères de cohésion et de séparation des clusters.

Afin d'utiliser le meilleur représentant de groupe possible pour l'application des mesures d'évaluation par reconstruction de liste aux systèmes expérimentés (le représentant est utilisé dans ces méthodes pour établir l'ordre des clusters ainsi que celui des documents à l'intérieur de chacun d'entre eux), nous avons expérimenté les différents modes de sélection présentés en section 2.4 (et même de nombreux compromis entre ces différents modes). Néanmoins, les résultats obtenus ne nous ont pas permis de déterminer un mode de sélection clairement supérieur aux autres. La sélection du document le plus proche de la requête dans chaque cluster semble, tel qu'observé dans [Leuski, 2001a], permettre l'obtention de bons résultats dans la plupart des cas pour l'ensemble des systèmes expérimentés. Par conséquent, dans l'ensemble des expérimentations que nous reportons dans la suite de ce mémoire, le représentant d'un cluster correspond à son document le plus proche de la requête.

Le tableau 8.4 présente les résultats obtenus par les quatre systèmes comparés selon les mesures présentées dans la section précédente. Alors que les résultats montrent que la méthode *Group-Average* obtient généralement le meilleur score selon la mesure *MK1* (qualité du cluster optimal, voir section 3.3.2), cette méthode semble produire un clustering des documents à partir duquel l'accès aux documents pertinents est plus difficile que celui proposé par la méthode *K-Means*. En effet, quel que soit le parcours emprunté, le score de précision moyenne que les systèmes utilisant cette méthode obtiennent est inférieur à celui présenté par les systèmes utilisant

Tit.	ZIFF				AP				WSJ				FR			
	G,N	K,N	G,Q	K,Q	G,N	K,N	G,Q	K,Q	G,N	K,N	G,Q	K,Q	G,N	K,N	G,Q	K,Q
NbP	1.10	**1.69**	1.19	1.65	0.82	**1.73**	1.01	1.37	0.96	**2.17**	1.08	1.98	1.75	**2.50**	1.75	2.30
MK1	**0.68**	0.72	0.70	0.73	**0.62**	0.66	0.63	0.67	**0.68**	0.75	0.70	0.77	**0.57**	0.67	**0.57**	0.68
PrP	0.61	**0.63**	0.60	0.62	0.61	**0.66**	0.60	0.62	0.62	**0.65**	0.62	0.64	0.70	**0.71**	0.69	0.70
PRO	0.41	**0.42**	**0.42**	0.41	**0.46**	0.41	0.43	0.42	0.52	**0.57**	0.52	0.55	**0.50**	0.49	0.48	0.47
LAR	0.35	**0.40**	0.34	0.39	0.32	**0.38**	0.33	0.34	0.42	**0.49**	0.40	0.48	0.47	**0.53**	0.47	0.52
LEU	0.43	**0.44**	0.43	0.42	**0.48**	0.44	0.44	0.43	0.55	**0.60**	0.53	0.57	**0.51**	**0.51**	0.50	0.49
OPT	0.52	**0.65**	0.52	0.62	0.54	**0.66**	0.53	0.62	0.63	**0.80**	0.62	0.78	0.70	**0.73**	0.69	0.71
PM1	0.38	**0.42**	0.36	0.40	0.36	**0.40**	0.36	0.36	0.47	**0.54**	0.45	0.53	0.49	**0.56**	0.48	0.53
PM2	0.42	**0.46**	0.42	0.43	**0.46**	**0.46**	0.45	0.44	0.54	**0.63**	0.51	0.60	0.50	**0.58**	0.50	0.56
PM3	0.43	**0.46**	0.40	0.42	0.45	**0.48**	0.46	0.46	0.53	**0.63**	0.51	0.61	0.51	**0.60**	0.51	0.59
Nar.	ZIFF				AP				WSJ				FR			
	G,N	K,N	G,Q	K,Q	G,N	K,N	G,Q	K,Q	G,N	K,N	G,Q	K,Q	G,N	K,N	G,Q	K,Q
NbP	2.25	2.61	2.32	**2.69**	1.84	**2.38**	1.93	2.36	1.85	2.58	1.97	**2.59**	2.49	3.02	2.71	**3.21**
MK1	**0.62**	0.68	**0.62**	0.68	0.59	0.64	**0.58**	0.64	**0.66**	0.72	0.67	0.71	**0.55**	0.62	0.56	0.62
PrP	0.64	0.66	0.64	**0.67**	0.64	**0.67**	0.65	**0.67**	0.64	**0.67**	0.66	**0.67**	0.70	0.73	0.70	**0.74**
PRO	**0.54**	0.51	0.53	0.50	**0.55**	0.53	**0.55**	0.52	0.62	0.61	**0.63**	0.60	**0.54**	0.52	0.52	0.52
LAR	0.50	**0.54**	0.51	**0.54**	0.46	0.51	0.46	**0.52**	0.49	0.54	0.49	**0.56**	0.48	0.56	0.49	**0.58**
LEU	**0.55**	0.53	**0.55**	0.52	0.56	0.55	**0.57**	0.55	0.65	0.64	**0.66**	0.63	0.56	0.56	0.56	**0.57**
OPT	0.68	**0.76**	0.69	0.75	0.72	0.77	0.72	**0.79**	0.74	0.82	0.76	**0.83**	0.74	0.78	0.74	**0.82**
PM1	0.52	**0.55**	0.52	**0.55**	0.49	0.52	0.49	**0.53**	0.55	0.58	0.55	**0.61**	0.52	0.63	0.54	**0.65**
PM2	0.55	**0.57**	**0.57**	**0.57**	0.55	0.58	0.56	**0.59**	0.65	0.69	0.67	**0.70**	0.57	0.65	0.57	**0.67**
PM3	0.55	0.59	0.56	**0.60**	0.54	0.58	0.56	**0.61**	0.66	0.68	0.67	**0.70**	0.56	0.65	0.58	**0.71**

TABLE 8.4 – Résultats des systèmes

la méthode *K-Means*[15]. Dans tous les cas, les différentes mesures que nous avons proposées permettent de conclure à une plus grande capacité à atteindre les documents pertinents à partir d'une liste de clusters proposée par un système utilisant la méthode *K-Means*.

Selon nous, la *Query Sensitive Similarity Measure* souffre du fait qu'elle tend à minorer la similarité entre documents partageant un grand nombre de termes n'apparaissant pas dans la requête, que ces documents partagent ou non un grand nombre de termes de la requête. De plus, un document pertinent contenant peu de termes de la requête ne peut pas, par l'utilisation d'une telle mesure, bénéficier de sa proximité avec d'autres documents pertinents, tel que cela pourrait être le cas avec des mesures de similarité classiques. Néanmoins, lorsque la requête est suffisamment longue, ces biais sont équilibrés par les bénéfices tirés de l'augmentation de la distance entre documents ne possédant pas de termes de la requête en commun (ce qui peut ne pas survenir lorsque la requête est courte puisque l'ensemble des documents considérés sont susceptibles de contenir la totalité des termes de la requête). Par

15. À l'exception de quelques cas où le parcours en profondeur et le parcours proposé dans [Leuski, 2001a], parcours qui favorisent les approches regroupant la plupart des documents pertinents, leur attribuent des scores supérieurs.

la considération des mesures d'évaluation existantes, aucune amélioration significative n'est cependant observée. L'augmentation de la capacité à regrouper l'ensemble des documents pertinents dans un même cluster, lorsque ceux-ci abordent le sujet de l'utilisateur sous un même angle, est contre-balancée par les pénalités que les évaluations attribuent à la mesure *QSSM* pour avoir conduit à la distribution de l'information pertinente dans plusieurs groupes lorsque la requête comporte plusieurs aspects bien distincts. Les approches d'évaluation proposées dans ce chapitre permettent de réparer cette injustice en ne favorisant pas un type de clustering par rapport à un autre, mais en déterminant les performances d'un système uniquement selon les facilités qu'il offre à un utilisateur pour trouver les informations qu'il recherche. Les approches d'évaluation que nous proposons, et tout particulièrement le parcours orienté par la proximité des documents pertinents $PM3$, permettent alors de mettre en valeur les bénéfices tirés de la prise en compte de la requête dans le processus de clustering.

8.4 Conclusion

L'application de techniques de clustering sur les résultats d'une recherche d'information a pour but d'en faire émerger les thématiques principales. L'objectif est de guider l'utilisateur dans sa recherche en lui présentant la structure de l'ensemble des textes retournés. Néanmoins, le niveau de diversité des textes considérés implique bien souvent un faible degré de finesse du clustering réalisé et certaines thématiques émergentes peuvent se trouver en forte déconnection avec la requête de l'utilisateur. La plupart des systèmes réalisant une catégorisation des résultats d'une recherche préliminaire y voient un effet bénéfique puisque cela permet de regrouper la plupart des textes pertinents dans un même cluster, donnant ainsi la possibilité à un utilisateur de filtrer les résultats retournés en ne parcourant que le cluster contenant les informations qui l'intéressent. Adoptant un point de vue différent, nous considérons ce phénomène comme largement négatif pour de multiples raisons, un tel regroupement de l'information pertinente ne permettant notamment pas de présenter à un utilisateur les différents aspects de sa requête et risquant alors de lui en restreindre la perception à un unique point de vue. Selon nous, une distribution de l'information pertinente sur l'ensemble des groupes formés peut s'avérer bien plus intéressante que sa concentration dans un unique cluster. Tout dépend du niveau d'accessibilité des textes pertinents à partir de la liste de descriptions de clusters présentée. Le fait de ne s'intéresser, pour l'évaluation d'un système de recherche réalisant une catégorisation des résultats, qu'au cluster contenant le meilleur ratio de textes pertinents induit un paradoxe important : si l'ensemble des clusters formés représentent des aspects bien distincts de la requête formulée, dans lesquels les textes pertinents sont facilement accessibles à partir des descriptions présentées, le système est jugé moins performant que si il avait rassemblé la majorité des textes pertinents dans un même cluster, se désintéressant alors des différences thématiques pouvant exis-

ter entre ces textes. Les mesures d'évaluation par reconstruction de listes ordonnées de documents permettent une comparaison des systèmes plus équitable puisqu'elles considèrent l'ensemble des clusters proposés à l'utilisateur. Néanmoins, nous nous sommes aperçus que les approches de reconstruction existantes présentaient une forte tendance à favoriser les systèmes regroupant l'ensemble des textes pertinents dans un même cluster. Nous avons alors cherché à établir des approches de reconstruction de listes qui rendent réellement compte de la capacité à atteindre les informations pertinentes à partir de la liste de descriptions de clusters présentée à l'utilisateur. À la lumière des approches proposées, qui semblent refléter efficacement le comportement d'un utilisateur réel face à une liste de clusters, nous avons mis en évidence les bénéfices potentiels résultant d'une prise en compte de la requête dans le processus de clustering. Bien que n'ayant pas été nécessairement conçues dans un tel but, les approches intégrant une considération de la requête dans leurs calculs de similarité semblent permettre, lorsque la requête est suffisamment large, d'en faire émerger les différents aspects en répartissant les documents traitant du sujet de la recherche sous des angles différents dans des clusters distincts. Ce chapitre a alors posé les bases du système que nous cherchons à concevoir, rendant compte des bénéfices potentiels qui pouvaient être tirés de la distribution de l'information pertinente, et définissant un certain nombre d'outils pouvant s'avérer utile pour son évaluation.

Chapitre 9

Présenter un aperçu de l'information pertinente

Le chapitre précédent ayant mis en évidence le fait qu'une prise en compte de la requête dans les calculs de similarité pouvait permettre, de manière indirecte, de distinguer certains aspects de l'information recherchée, nous proposons ici d'agir directement sur le processus de clustering afin d'organiser les différents clusters autour du sujet recherché par l'utilisateur et ainsi obtenir une liste de représentants de cluster constituant un réel aperçu de l'information pertinente qu'il est possible de trouver dans le corpus considéré. Ce chapitre présente et évalue dans un premier temps l'algorithme de clustering proposé, puis l'applique au niveau des segments thématiques afin de composer un document qui puisse permettre à un utilisateur d'appréhender plus aisément la structure de l'information qu'il recherche.

Sommaire

9.1 Un clustering multi-objectifs orienté requête 214
9.2 Évaluation du système 217
9.2.1 Influence du critère de proximité des groupes avec la requête . . 218
9.2.2 Influence du nombre de documents considérés 221
9.2.3 Influence du nombre de groupes produits 222
9.2.4 Détermination du nombre de groupes optimal 223
9.3 Application aux segments thématiques 224
9.3.1 Influence de la segmentation sur l'organisation des clusters . . . 224
9.3.2 Que présenter à l'utilisateur ? 226
9.4 Conclusion 227

9.1 Un clustering multi-objectifs orienté requête

Au vu des résultats présentés en section 8.3, le fait de considérer la requête lors des calculs de similarité entre documents semble améliorer l'accès aux informations pertinentes. La mesure *QSSM*, intégrant dans le calcul de la similarité entre deux documents une prise en compte de leur proximité commune à la requête, renforce les relations entre documents partageant les mêmes termes de la requête et affaiblit celles des documents se rapprochant de la requête sur des termes différents. Bien que l'objectif énoncé lors de la définition de cette mesure était de permettre un meilleur regroupement des documents pertinents, il s'avère qu'elle permet parfois, lorsque le sujet de la requête est suffisamment large, de distinguer différents aspects de l'information recherchée par l'utilisateur. Néanmoins, cette mesure présente selon nous un certain nombre de limites, concernant notamment le fait que les termes de la requête ont un très fort impact sur la qualité du partitionnement obtenu (voir section 8.3). Par ailleurs, ce type d'approche ne permet pas un réel contrôle de la distribution de l'information pertinente. Selon les similarités entre documents, il arrive que des groupes de documents pertinents se distinguent par leur affectation à des clusters différents mais dans la majorité des cas, les documents pertinents tendent à se regrouper dans un même cluster.

Afin d'agir plus directement sur la distribution de l'information pertinente, nous proposons de nous concentrer sur le processus de clustering lui même plutôt que de modifier l'espace des similarités tel que réalisé par la plupart des approches de clustering orienté requête. Dans un premier temps, des expérimentations concernant la reformation de clusters autour des représentants de chaque groupe (chaque document est alors ré-affecté au cluster du représentant de groupe qui lui est le plus proche) ont été menées avec de nombreuses heuristiques de sélection des documents représentants (avec un intérêt particulier pour les représentants sélectionnés entre le centre du cluster et le document le plus proche de la requête) dans le but de déterminer si un tel processus pouvait conduire à un meilleur partitionnement de l'information pertinente que les approches de clustering orienté requête existantes. Ces expérimentations n'ont cependant permis d'observer aucune amélioration significative des résultats (et même une légère baisse des performances). Cela peut certainement s'expliquer par le fait que les documents représentants sont choisis dans des groupes initialement formés sans aucune considération de la requête, certains des représentants pouvant alors être relativement déconnectés du besoin de l'utilisateur. Cette observation nous a conduit à proposer une méthode de clustering dans laquelle les clusters sont formés de la même manière, autour de quelques documents choisis (appelés ci-après les leaders), mais où la sélection de ces documents ne dépend pas de groupes formés par un partitionnement initial. Cette méthode cherche à guider la formation des clusters selon le contexte de la recherche d'information en prenant en compte un critère de proximité des documents leaders avec la requête. Afin d'obtenir des clusters représentant différents aspects de l'information

pertinente, chaque groupe doit en effet être formé autour d'un document fortement connecté avec le sujet de la recherche.

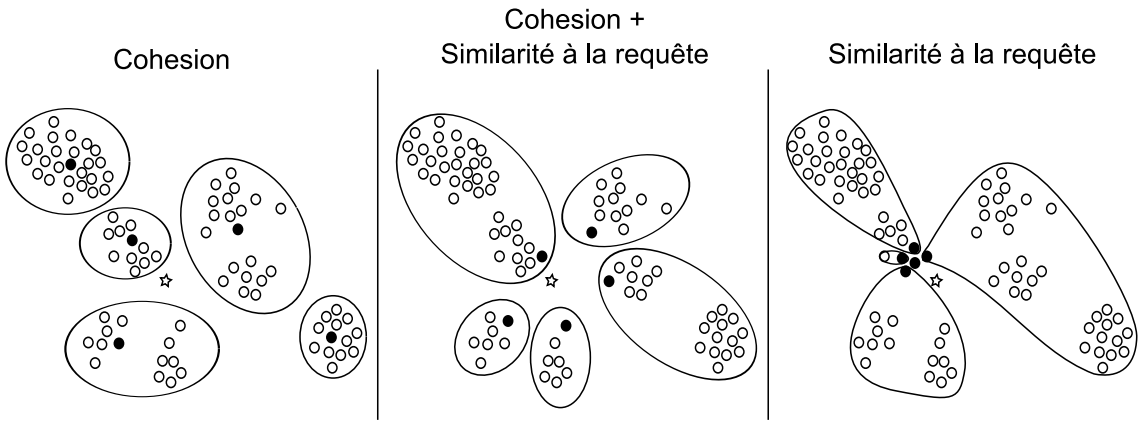

FIGURE 9.1 – Effets des critères optimisés sur les groupes de documents formés.

La figure 9.1 illustre trois exemples de clusterings réalisés sur un même ensemble de documents selon différents critères : les documents sont représentés par des cercles et la requête par une étoile, les distances entre objets correspondent à leur dissimilarité thématique et les disques noirs représentent les leaders de chaque groupe (documents autour desquels les clusters ont été formés, chaque document ayant été affecté au cluster du leader qui lui est le plus proche). Cette figure met en évidence le fait qu'un clustering optimisant un unique critère de cohésion lors de la sélection des leaders, tel que celui présenté à gauche de la figure, conduit bien souvent à l'obtention de groupes largement déconnectés du besoin de l'utilisateur (et d'un groupe contenant les différents aspects de l'information pertinente). À l'inverse, le fait d'optimiser un unique critère de proximité des leaders à la requête, tel que le clustering à droite de la figure, conduit à l'obtention de clusters sans grande signification, ne représentant pas des thématiques particulières. Selon la figure 9.1, le meilleur clustering est obtenu par la réalisation d'un compromis entre ces deux critères de cohésion et de proximité à la requête. L'optimisation conjointe de ces deux critères permet en effet d'obtenir des clusters organisés autour de la requête, qui sont alors supposés mieux représenter les différentes thématiques relatives au sujet qui intéresse l'utilisateur. Par conséquent, plutôt que de ne considérer qu'un seul critère de cohésion des groupes (ou de similarité des documents avec le leader de leur groupe), tel que c'est le cas dans la plupart des approches de clustering existantes, la méthode que nous proposons ici prend également en compte un critère de proximité des leaders avec la requête. Puisque la cohésion des groupes a tendance à diminuer lorsque la proximité moyenne des leaders avec la requête augmente, nous nous trouvons face à un problème multi-objectifs. Cette observation nous a alors conduit, tel que nous l'avons fait pour la segmentation thématique de textes (section 5.4.2), à

employer l'algorithme génétique *Strength Pareto Evolutionary Algorithm*. Cet algorithme dédié à l'optimisation multi-objectifs ayant été présenté en détail en section 4.3.2, nous n'explorons ici que les aspects et détails spécifiques à notre problème de clustering orienté requête.

Dans [Maulik and Bandyopadhyay, 2000], les auteurs ont appliqué un algorithme génétique au problème du clustering de données pour surmonter la principale limitation de l'algorithme *K-means*, bien connu pour produire un partitionnement fortement dépendant des centres de groupes initialement choisis. L'objectif de cet algorithme est alors de trouver l'ensemble de centres de clusters permettant de maximiser un critère de cohésion moyenne. Seuls les centres sélectionnés sont représentés dans le codage des individus, les différents clusters découlant directement des ces centres, chaque document étant finalement affecté au groupe du centre qui lui est le plus proche. Pour un espace de données à n dimensions, la longueur d'un chromosome est alors égale à $n \times k$, où k représente le nombre de groupes à produire (et donc le nombre de centres à représenter). Dans un contexte de recherche d'information, cette représentation conduit à un espace de recherche bien trop grand puisque le nombre de dimensions de notre espace de données correspond au nombre de termes du corpus. La représentation que nous utilisons est inspirée de celle présentée dans [Maulik and Bandyopadhyay, 2000] mais, plutôt que de définir les points centraux des groupes, les gènes de nos individus correspondent aux documents leaders des clusters. Les clusters d'un individu sont finalement obtenus en affectant chaque document au leader qui lui est le plus proche. Chaque individu manipulé par notre algorithme est alors représenté par l'ensemble des index de ses k leaders. Étant donné un ensemble $\mathcal{D}$ de n documents à partitionner en k clusters, les individus sont alors des vecteurs $\vec{x}$ de k éléments x_j ; x_j correspondant au j-ième leader de l'individu. Chaque individu $\vec{x}$ est évalué selon deux critères, le critère $C(\vec{x})$ correspondant à la proximité moyenne des documents avec le leader qui leur est le plus proche (en terme de la mesure de similarité inter-textes utilisée, dans nos expérimentations *NCosine* ou *QSSM*) et le critère $S(\vec{x})$ correspondant à la similarité moyenne des leaders avec la requête R (en terme de le mesure *NCosine* définie en section 6.2.2) :

$$C(\vec{x}) = \sum_{i=1}^{n} \max_{j \in \{1,\ldots,k\}} Sim(D_i, D_{x_j})/n \tag{9.1}$$

$$S(\vec{x}) = \sum_{j=1}^{k} Sim(D_{x_j}, R)/k \tag{9.2}$$

Notre problème d'optimisation revient alors à approcher l'ensemble des maximi-

seurs $\mathcal{M}$ des deux objectifs $C(\vec{x})$ et $S(\vec{x})$:

$$\begin{aligned}\mathcal{M} = \Big\{ & \vec{x} \in \{1,\ldots,n\}^k | \ \forall \vec{x'} \in \{1,\ldots,n\}^k, \\ & \vec{x} \neq \vec{x'} \Rightarrow \Big(C(\vec{x}) > C(\vec{x'}) \ \vee \ S(\vec{x}) > S(\vec{x'}) \\ & \qquad \vee \ \big(C(\vec{x}) = C(\vec{x'}) \ \wedge \ S(\vec{x}) = S(\vec{x'}) \big) \Big) \Big\}\end{aligned} \tag{9.3}$$

Les algorithmes génétiques ont tendance à converger plus facilement vers des solutions "optimales" lorsque les individus de la population initiale sont déjà des solutions relativement acceptables [Goldberg, 1989]. Par conséquent, nous initialisons la population de notre algorithme avec des jeux de leaders correspondant à des ensembles de représentants de clusters obtenus par l'algorithme *K-Means*. Les différences entre les individus résident dans le mode de sélection de ces représentants : selon l'individu cible, les représentants sont sélectionnés dans chaque cluster selon une fonction d'agrégation des deux critères différente (critères de similarité avec la requête et de similarité moyenne avec les autres membres du groupe). Ce processus de génération permet à l'algorithme de démarrer avec une population hétérogène de solutions relativement satisfaisantes.

Trois opérateurs sont utilisés dans le processus d'évolution : la sélection, le croisement et la mutation. À chaque génération, l'algorithme sélectionne N individus et en produit N nouveaux afin de maintenir une population de taille fixe. Les expérimentations que nous avons conduites ont montré qu'un nombre d'individus de $N = 10$ permet une bonne convergence de l'algorithme dans la majorité des cas. Les individus sont sélectionnés dans les deux populations $\bar{P} \cup P_t$ par sélection proportionnelle selon leur valeur d'adaptation [1] (voir section 4.2). Parmi les individus sélectionnés, tant que le nombre de N nouveaux individus n'est pas atteint, deux individus sont choisis aléatoirement pour en produire un nouveau par croisement uniforme : le j-ième leader du nouvel individu est choisi équiprobablement dans l'un des deux parents (tout en évitant de choisir deux fois le même leader). Un opérateur de mutation remplaçant un leader par un autre document (qui n'appartient pas déjà au jeu de leaders de l'individu) est appliqué au nouvel individu avec une probabilité de 0.5 (taux de mutation qui nous a paru produire les meilleurs résultats). Lorsque le nombre d'individus appartenant à l'archive dépasse les $3 \times N$ individus, une réduction de l'archive est opérée de la même manière que dans notre algorithme de segmentation thématique (voir section 5.4.2).

9.2 Évaluation du système

À la fin du processus d'évolution, l'archive $\bar{P}$ (voir section 4.3.2) constitue un ensemble de solutions de clustering potentielles. L'algorithme doit alors en extraire

1. Les valeurs d'adaptations des individus sont calculées avec les formules de Zitzler, formules 4.4 et 4.5, selon nos deux objectifs de cohésion et de proximité des leaders à la requête.

une solution pour présenter un partitionnement à l'utilisateur. Cette extraction est réalisée par agrégation des scores des deux objectifs. Le clustering obtenu est fortement dépendant de la pondération que l'on applique aux deux critères : un poids trop important attribué au critère de cohésion risque de conduire à l'obtention de clusters déconnectés du besoin de l'utilisateur, un poids trop important attribué à la similarité des leaders avec la requête risque de conduire à l'obtention de clusters dont les éléments n'ont rien à voir les uns avec les autres. Afin de répartir équitablement les scores des deux critères et de simplifier ainsi la tâche de pondération, nous identifions, pour chacun des deux critères, la valeur minimale et la valeur maximale prises par des individus de l'archive et nous utilisons ces valeurs pour étaler les scores de tous les individus dans les régions $[0, 1] \times [0, 1]$. De la même manière que réalisé dans [Handl and Knowles, 2007], nous considérons finalement la racine carré des scores obtenus pour chaque objectif afin d'écarter les solutions et donner un poids plus important aux compromis situés au centre du front identifié. Le front étant alors relativement bien équilibré, une simple agrégation (sans pondération) des scores des individus permet d'extraire la solution (celle obtenant le meilleur score d'agrégation) optimisant équitablement les deux critères considérés.

9.2.1 Influence du critère de proximité des groupes avec la requête

Afin d'évaluer le bénéfice résultant de la considération de la requête, nous décrivons dans cette section des expérimentations réalisées avec des solutions extraites selon deux pondérations différentes des objectifs : dans les approches C, N et C, Q (C pour Cohésion, N et Q correspondent à la mesure de similarité entre documents utilisée par l'algorithme, *NCosine* ou *QSSM*), la solution est extraite de l'archive en attribuant un poids nul au critère de similarité des leaders avec la requête (ce qui revient à une méthode de clustering classique considérant un unique critère de cohésion des groupes), dans les approches M, N et M, Q (M pour Multi-objectifs), la solution est extraite en attribuant un poids équivalent aux deux critères (d'autres pondérations ont été expérimentées mais n'ont pas conduit à de meilleurs résultats).

Les résultats expérimentaux présentés dans cette section concernent des partitionnements en cinq clusters des 50 premiers documents retournés par une recherche initiale [2]. Les expérimentations sont réalisées sur les corpus présentés en section 7.2.2 avec les requêtes 1 à 50 de *TREC*. Par ailleurs, l'algorithme génétique est stoppé, pour l'ensemble des approches, au bout de 100 générations, ce qui permet d'obtenir une solution satisfaisante en un temps raisonnable (autour de 5 secondes sur un *Pentium 4, 3GHz PC*).

2. Étant donné l'aspect stochastique de la méthode, nous avons relancé le processus d'évolution 10 fois par requête pour obtenir un score moyen plus représentatif des performances réelles de la méthode. Les résultats reportés ici correspondent alors à ces scores moyens obtenus. Par ailleurs, nous n'avons pas observé, pour une requête et une mesure donnée, d'écart de score supérieur à 0.01 pour l'approche M, N, ce qui paraît vraiment faible au regard des écarts de score entre les différentes méthodes.

		ZIFF			AP			WSJ			FR		
		Co	*Sep*	*EcD*	*Co*	*Sep*	*EcD*	*Co*	*Sep*	*EcD*	*Co*	*Sep*	*EcD*
Tit	C, N	0.57	0.46	7.76	0.58	0.45	6.08	0.58	0.45	6.23	0.67	0.41	5.87
	M, N	0.57	0.46	7.21	0.58	0.45	7.49	0.58	0.45	6.80	0.66	0.42	10.66
	C, Q	0.56	0.46	8.64	0.57	0.45	9.41	0.57	0.46	9.35	0.65	0.42	7.12
	M, Q	0.56	0.45	7.18	0.57	0.45	7.83	0.57	0.46	8.37	0.66	0.40	8.66
Nar	C, N	0.58	0.45	6.59	0.59	0.45	5.96	0.59	0.45	6.35	0.64	0.44	5.48
	M, N	0.58	0.45	6.28	0.59	0.45	6.55	0.59	0.45	6.59	0.63	0.45	7.94
	C, Q	0.57	0.45	7.93	0.59	0.45	5.96	0.58	0.45	7.79	0.64	0.44	5.72
	C, Q	0.57	0.45	6.97	0.59	0.45	6.47	0.58	0.45	6.98	0.64	0.44	7.01

TABLE 9.1 – Statistiques des partitionnements produits

Le tableau 9.1 présente les statistiques des clusterings produits par les différentes approches [3]. Selon les résultats présentés, on peut noter que les clusterings proposés par l'approche C, N présentent de meilleurs degrés de cohésion et de séparation des clusters que ceux produits par les méthodes *K-Means* et *Group-Average* (voir table 8.3). Le fait de considérer un critère de similarité des leaders avec la requête (M, N) ne semble pas affecter significativement les scores de cohésion et de séparation des clusters (contrairement à la prise en compte du contexte réalisée par la mesure *QSSM*).

Le tableau 9.2 présente les performances des différents systèmes (les notations des mesures sont données dans la table 8.1) avec des documents ordonnés dans chaque cluster selon leur similarité avec le représentant de leur groupe [4] et des clusters ordonnés selon la similarité du représentant avec la requête (ces deux classements sont réalisés selon la mesure *NCosine*). Selon les résultats présentés, on peut observer le gain réalisé par la prise en compte du contexte lors de l'utilisation de requêtes longues (*Nar*). Alors que le score du parcours en profondeur diminue légèrement lorsque l'on considère un critère de proximité des leaders à la requête (M, N), le score du parcours en largeur est fortement amélioré. La prise en compte du contexte par l'algorithme que nous proposons paraît avoir un impact très positif des résultats obtenus selon les parcours orientés par la pertinence des documents $PM2$ et par la proximité des documents pertinents $PM3$. Un test de Student a montré significatives, avec un degré de confiance de 99%, les differences entre les résultats obtenus sur ces parcours par les approches C, N et M, N. Notre approche paraît réaliser une bien meilleure distribution de l'information pertinente que la prise en compte

3. Les notations des mesures sont les mêmes que dans la table 8.3 : *Co* correspond à la cohésion moyenne des groupes (formule 2.2), *Sep* à leur séparation (formule 2.3) et *EcD* à l'écart type entre les tailles de clusters. Les critères de cohésion et de séparation des clusters sont calculés selon la mesure *NCosine*.

4. Afin d'être à même de rapprocher les résultats reportés par ce tableau et ceux présentés dans le tableau 8.4, le représentant d'un groupe correspond à son document le plus proche de la requête. Il est à noter que des résultats relativement similaires ont été observés, pour les approches M, N et M, Q, en utilisant directement les leaders comme représentants de groupe.

Tit.	ZIFF				AP				WSJ				FR			
	C,N	M,N	C,Q	M,Q	C,N	M,N	C,Q	M,Q	C,N	M,N	C,Q	M,Q	C,N	M,N	C,Q	M,Q
NbP	1.52	**1.93**	1.60	1.84	1.71	1.89	1.65	**1.90**	2.31	2.41	2.14	**2.55**	2.45	**3.25**	2.30	**3.25**
*MK*1	**0.71**	0.73	0.72	0.73	0.65	0.65	**0.64**	0.65	**0.76**	0.77	0.77	0.77	**0.67**	0.69	0.69	0.72
PrP	0.64	**0.66**	0.62	0.64	0.65	**0.67**	0.64	0.64	0.65	**0.66**	0.65	**0.66**	**0.74**	**0.74**	0.70	0.71
PRO	**0.43**	0.42	0.42	0.42	**0.40**	**0.40**	**0.40**	0.37	**0.56**	0.53	0.53	0.53	**0.52**	**0.52**	0.47	0.46
LAR	0.40	**0.42**	0.40	0.41	0.38	**0.39**	0.36	**0.39**	0.50	**0.53**	0.49	0.50	0.54	**0.57**	0.52	0.51
LEU	**0.45**	**0.45**	0.44	0.44	**0.43**	**0.43**	0.42	0.40	**0.59**	0.56	0.56	0.56	0.53	**0.54**	0.50	0.48
OPT	0.67	**0.70**	0.64	0.64	**0.66**	**0.66**	0.61	0.64	**0.80**	0.79	0.77	0.77	0.77	**0.78**	0.72	0.73
*PM*1	0.42	**0.44**	0.42	0.43	**0.40**	**0.40**	0.38	0.39	0.54	**0.56**	0.52	0.53	0.58	**0.62**	0.54	0.52
*PM*2	0.45	**0.47**	0.44	0.45	0.45	**0.46**	0.44	0.43	0.63	**0.64**	0.60	0.60	0.61	**0.64**	0.56	0.55
*PM*3	0.45	**0.48**	0.43	0.46	0.49	**0.50**	0.45	0.45	0.63	**0.65**	0.60	0.61	0.63	**0.67**	0.58	0.57
Nar.	ZIFF				AP				WSJ				FR			
	C,N	M,N	C,Q	M,Q	C,N	M,N	C,Q	M,Q	C,N	M,N	C,Q	M,Q	C,N	M,N	C,Q	M,Q
NbP	2.52	3.01	2.80	**3.16**	2.34	**2.82**	2.34	**2.82**	2.57	**3.14**	2.61	3.12	3.06	3.57	3.12	**3.63**
*MK*1	**0.67**	0.69	0.68	0.69	0.64	0.65	**0.63**	0.64	0.73	0.75	**0.70**	0.75	0.62	0.64	**0.59**	0.61
PrP	0.66	0.67	0.66	**0.68**	0.67	0.68	**0.69**	**0.69**	0.67	**0.69**	0.67	**0.69**	0.74	**0.76**	0.74	0.75
PRO	0.52	0.51	**0.53**	**0.53**	**0.54**	**0.54**	**0.54**	0.53	0.59	0.59	**0.60**	**0.60**	**0.54**	**0.54**	**0.54**	**0.54**
LAR	0.54	**0.56**	0.55	**0.56**	0.52	0.57	0.54	**0.58**	0.57	0.60	0.59	**0.62**	0.56	**0.60**	0.57	0.59
LEU	0.54	0.53	**0.55**	**0.55**	**0.56**	**0.56**	**0.56**	**0.56**	0.63	**0.64**	**0.64**	**0.64**	0.57	**0.58**	0.57	**0.58**
OPT	0.76	0.78	0.77	**0.79**	0.79	**0.82**	0.81	**0.82**	0.82	0.85	0.84	**0.87**	0.80	**0.84**	0.81	**0.84**
*PM*1	0.55	0.57	0.56	**0.58**	0.53	0.58	0.55	**0.59**	0.61	0.64	0.63	**0.66**	0.63	**0.67**	0.64	0.66
*PM*2	0.57	0.60	0.59	**0.61**	0.60	0.65	0.62	**0.66**	0.69	0.73	0.72	**0.75**	0.66	**0.70**	0.67	**0.70**
*PM*3	0.60	0.64	0.62	**0.65**	0.62	**0.67**	0.63	**0.67**	0.69	0.74	0.71	**0.75**	0.70	**0.73**	0.71	**0.73**

TABLE 9.2 – Résultats des systèmes

de la requête réalisée par la mesure *QSSM* puisque les résultats de M, N sont bien meilleurs que ceux obtenus par C, Q sur l'ensemble des corpus. Par ailleurs, avec ces requêtes longues, l'utilisation de la mesure *QSSM* dans notre algorithme de clustering (M, Q) permet d'améliorer encore un peu les résultats obtenus.

Avec les requêtes courtes (*Tit*), une plus grande similarité avec la requête pour un document des 50 premiers retournés par la recherche initiale n'implique pas nécessairement un plus fort potentiel de pertinence puisque l'ensemble de ces 50 documents retournés sont susceptibles de contenir la totalité des termes de la requête. Par conséquent, le fait de considérer un critère de similarité des leaders avec la requête ne permet pas d'observer de telles améliorations des résultats qu'avec les requêtes longues. Néanmoins, contrairement aux observations réalisées avec la mesure *QSSM*, les performances des systèmes ne sont pas dégradées par notre mode de considération du contexte lors de l'utilisation de requêtes courtes (elles sont même souvent légèrement améliorées, particulièrement sur les corpus ZIFF et FR où les différences avec les résultats de l'approche C, N ont été montrées significatives par un test de Student).

9.2.2 Influence du nombre de documents considérés

Des expérimentations additionnelles, reportées par les graphes de la figure 9.2, ont concerné l'influence du nombre de documents considérés sur les performances de notre approche. Pour ce faire, nous avons comparé les scores obtenus par le parcours $PM3$ orienté par la proximité des documents pertinents, pour les deux approches C, N (sans prise en compte du contexte) et M (utilisant un critère de proximité des leaders avec la requête), sur des clusterings des 50, 100, 200 et 300 premiers documents retournés par un système initial [5].

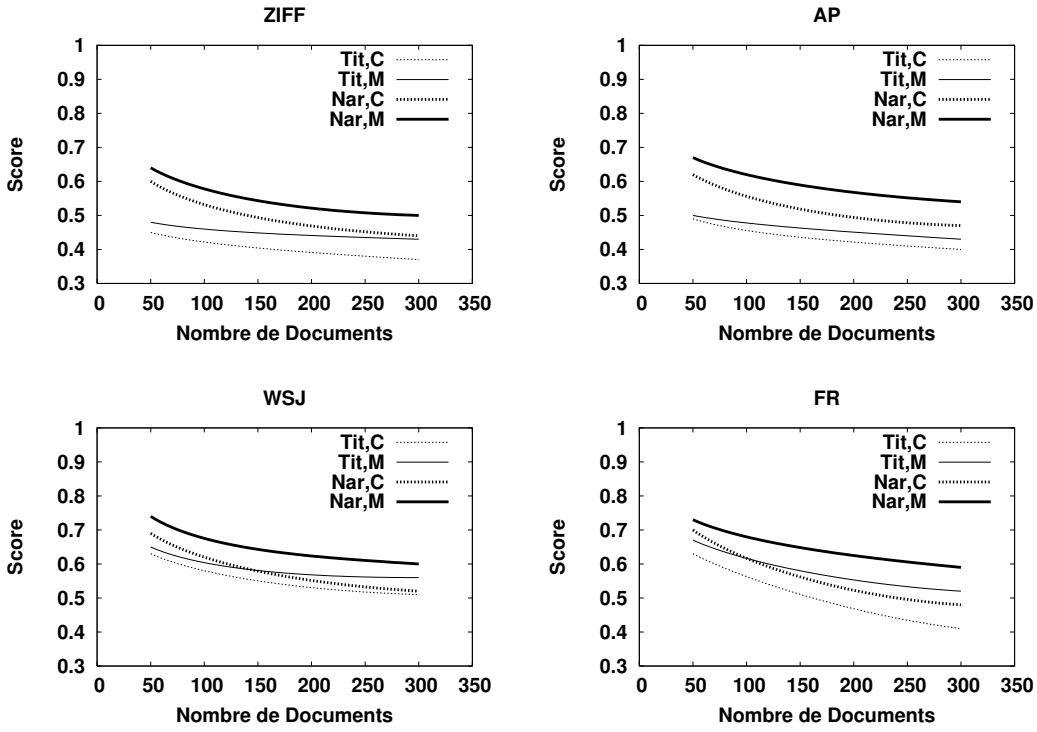

FIGURE 9.2 – Influence du nombre de documents considérés.

Les résultats obtenus montrent que le gain réalisé par la prise en compte d'un critère de proximité des clusters avec la requête tend à s'amplifier lorsque le nombre de documents considérés augmente. Cela peut s'expliquer par le fait que plus le nombre de documents considérés est important, plus il y a des chances pour que certains d'entre eux soient éloignés de la requête et donc, plus les clusters produits

5. Sur la figure 9.2, la courbe Tit, C correspond aux résultats obtenus avec l'approche C, N en considérant des requêtes courtes (*Title*), la courbe Tit, M aux résultats obtenus avec M, N en considérant les mêmes requêtes *Title*, la courbe Nar, C à ceux obtenus avec C, N sur les requêtes longues (*Narrative*) et la courbe Nar, M à ceux obtenus avec M, N sur les mêmes requêtes *Narrative*. Le nombre de clusters produits est toujours de 5.

par une méthode ne considérant pas le contexte de la recherche risquent d'être déconnectés du sujet qui intéresse l'utilisateur.

9.2.3 Influence du nombre de groupes produits

Dans l'ensemble des expérimentations reportées dans cette section, le nombre de clusters à produire était arbitrairement fixé à 5. La figure 9.3 présente les résultats obtenus, avec la mesure d'évaluation $PM3$, par nos deux approches C, N et M, N produisant différents nombres de clusters entre 3 et 7. Le nombre de documents considérés est de 50.

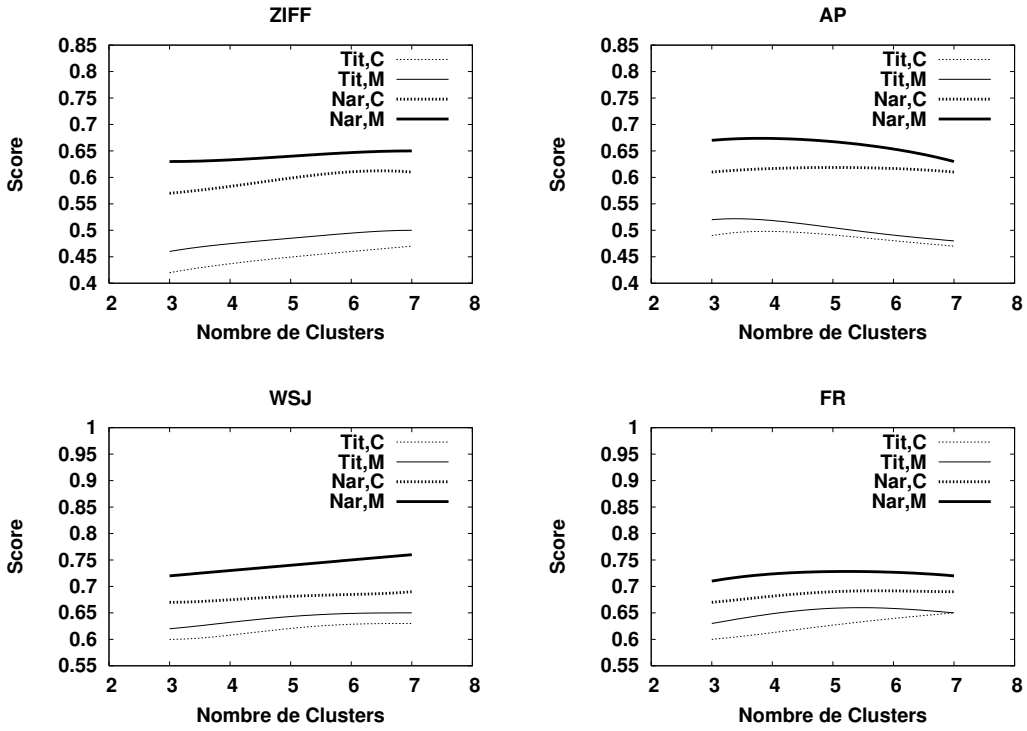

FIGURE 9.3 – Influence du nombre de clusters produits

Les résultats présentés montrent que les observations réalisées précédemment (à savoir le fait que la considération d'un critère de proximité des clusters avec la requête permet une amélioration de l'accès à l'information) n'étaient pas dues à un nombre de clusters plus adapté à l'approche M, N qu'à l'approche C, N. Ce gain de performances est en effet observé avec tous les nombres de clusters testés.

9.2.4 Détermination du nombre de groupes optimal

Le nombre optimal de clusters étant susceptible de varier selon les requêtes et les corpus considérés, nous proposons ici d'adapter notre algorithme afin qu'il puisse déterminer automatiquement le nombre de groupes qu'il doit produire. Le principe est d'obtenir un front de compromis présentant des nombres de groupes différents, pour en extraire la meilleure solution selon une fonction d'agrégation des objectifs. L'opérateur de croisement de notre algorithme doit être modifié pour permettre des variations des nombres de clusters représentés par les individus (l'initialisation de la population et les opérateurs de sélection et de mutation restent inchangés) : chaque leader des deux parents est alors sélectionné pour apparaître dans l'ensemble des leaders du nouvel individu avec une probabilité de 50% (en évitant de sélectionner deux fois le même leader). Avec un tel opérateur, le nombre de leader du nouvel individu n'est pas nécessairement le même que celui de ses parents. L'ensemble de compromis finalement obtenu est donc susceptible de contenir des individus aux nombres de clusters différents. Du fait de l'influence du nombre de clusters sur les deux objectifs, les solutions sont relativement réparties sur le front de compromis selon le nombre de clusters pour lesquelles elles codent : alors qu'une solution codant pour un faible nombre de clusters tend à présenter un score de similarité des leaders avec la requête élevé, une solution codant pour un grand nombre de clusters tend à présenter un meilleur score de cohésion. Tel que mentionné dans [Handl and Knowles, 2007], la structure de l'ensemble de données a tendance à se refléter sur la forme du front. C'est à dire que le nombre de clusters optimal est présenté par la solution réalisant le meilleur compromis des deux objectifs considérés. Cette solution est extraite du front de la même manière que précédemment.

Le tableau 9.3 présente les résultats obtenus sur chaque corpus par des individus extraits de fronts de compromis contenant différents nombres de clusters (M_{auto}). Le

		ZIFF				AP				WSJ				FR			
		PRO	*LAR*	*PM2*	*PM3*	*PRO*	*LAR*	*PM2*	*PM3*	*PRO*	*LAR*	*PM2*	*PM3*	*PRO*	*LAR*	*PM2*	*PM3*
Tit	M_{auto}	0.43	0.44	0.53	0.54	0.42	0.43	0.50	0.52	0.59	0.49	0.65	0.64	0.57	0.56	0.70	0.72
	M_{k-1}	0.42	0.44	0.53	0.53	0.42	0.47	0.51	0.54	0.54	0.53	0.66	0.66	0.54	0.57	0.68	0.71
	M_k	0.45	0.47	0.54	0.56	0.44	0.46	0.52	0.54	0.56	0.53	0.66	0.67	0.56	0.58	0.70	0.73
	M_{k+1}	0.40	0.42	0.52	0.51	0.48	0.44	0.51	0.53	0.59	0.51	0.66	0.66	0.56	0.56	0.69	0.71
Nar	M_{auto}	0.54	0.60	0.66	0.69	0.55	0.58	0.68	0.69	0.59	0.58	0.70	0.72	0.56	0.63	0.72	0.76
	M_{k-1}	0.50	0.64	0.68	0.70	0.53	0.59	0.68	0.69	0.60	0.59	0.72	0.73	0.54	0.61	0.71	0.74
	M_k	0.54	0.63	0.69	0.72	0.56	0.59	0.70	0.72	0.60	0.64	0.77	0.77	0.57	0.63	0.74	0.77
	M_{k+1}	0.56	0.60	0.66	0.69	0.56	0.58	0.68	0.70	0.59	0.60	0.74	0.74	0.56	0.60	0.72	0.75

Table 9.3 – Estimation du nombre de clusters optimal

nombre moyen de clusters produits est égal à 6.9 pour les requêtes *Title* et 6.4 pour les requêtes *Narrative* sur le corpus *ZIFF*, à 4.2 et 4.9 sur *AP*, à 7.5 et 6.8 sur *WSJ* et à 6.3 et 6.7 sur *FR*. Les expérimentations reportées ici montrent tout d'abord

que le nombre optimal de clusters à produire varie effectivement selon la requête considérée et que notre algorithme s'en approche efficacement puisque les résultats sont pour la plupart significativement supérieurs à ceux présentés dans la table 9.2 où le nombre de clusters à produire était arbitrairement fixé à 5 pour l'ensemble des requêtes. Cette propension à approximer efficacement le nombre optimal de clusters à produire est souligné par le fait que l'utilisation du nombre estimé pour fixer le nombre de clusters de la version originale de l'algorithme M_k conduit à de meilleurs résultats que d'utiliser le nombre de clusters directement supérieur M_{k+1} ou directement inférieur M_{k-1}. Le fait que l'utilisation du nombre de clusters estimé pour fixer le nombre de clusters à produire dans la version originale de l'algorithme M_k conduise à de meilleurs résultats que ceux obtenus par les clusterings directement extraits du front de solutions proposé par l'approche M_{auto} peut s'expliquer par la plus grande taille de l'espace de recherche qu'implique la variabilité du nombre de clusters. Ce problème peut être résolu en cherchant à améliorer les opérateurs génétiques ou tout simplement en augmentant le nombre de générations du processus d'évolution. Alternativement, il est possible d'envisager que l'algorithme travaille avec un nombre de clusters variable pendant un nombre de générations donné, pour estimer un nombre de clusters auquel il se tient sur les générations suivantes. Quoi qu'il en soit, les expérimentations reportées dans cette section montrent que l'estimation du nombre optimal de clusters, et par là le nombre de représentants à présenter à l'utilisateur, peut être réalisée de manière efficace par l'approche que nous proposons ici.

9.3 Application aux segments thématiques

Le processus établi dans ce chapitre cherche à distribuer les documents pertinents sur l'ensemble des groupes produits malgré leur tendance naturelle à se regrouper dans un même cluster. La liste de représentants qui en résulte est alors bien plus informative, chacun des documents initialement présentés à l'utilisateur étant susceptible de représenter un aspect intéressant du sujet. Nous appliquons ici ce processus de clustering aux segments thématiques des documents pour construire un document composite qui puisse être utilisé par un utilisateur pour appréhender aisément la structure de l'information qu'il recherche.

9.3.1 Influence de la segmentation sur l'organisation des clusters

Au chapitre 7, nous nous sommes aperçus que l'individualisation des segments thématiques des documents pouvait permettre un meilleur regroupement des documents pertinents dans un même cluster. Nous proposons alors dans un premier temps d'en évaluer l'impact sur les groupes de documents formés par notre algorithme de clustering orienté requête. Ainsi, à l'instar de l'approche H présentée dans le chapitre 7, nous réalisons un clustering préliminaire des segments thématiques des

Tit.	ZIFF		AP		WSJ		FR	
	C, N	M, N	C, N	M, N	C, N	M, N	C, N	M, N
$MK1$	0.68	0.71	0.64	0.65	0.72	0.75	0.62	0.65
PrP	0.63	0.66	0.65	0.67	0.64	0.66	0.72	0.75
PRO	0.44	0.43	0.40	0.40	0.58	0.54	0.55	0.54
LAR	0.41	0.45	0.38	0.40	0.49	0.55	0.55	0.59
OPT	0.69	0.72	0.66	0.67	0.80	0.81	0.79	0.82
$PM2$	0.48	0.50	0.46	0.48	0.64	0.65	0.65	0.67
$PM3$	0.49	0.53	0.49	0.51	0.63	0.67	0.67	0.70
Nar.	ZIFF		AP		WSJ		FR	
	C, N	M, N	C, N	M, N	C, N	M, N	C, N	M, N
$MK1$	0.64	0.68	0.64	0.65	0.69	0.73	0.58	0.63
PrP	0.65	0.68	0.67	0.68	0.66	0.70	0.74	0.77
PRO	0.54	0.52	0.54	0.53	0.60	0.59	0.56	0.54
LAR	0.54	0.60	0.51	0.58	0.57	0.62	0.56	0.62
OPT	0.78	0.81	0.78	0.82	0.83	0.87	0.81	0.85
$PM2$	0.61	0.65	0.59	0.65	0.70	0.74	0.70	0.72
$PM3$	0.63	0.68	0.61	0.67	0.70	0.75	0.72	0.75

TABLE 9.4 – Clusterings de documents déduits de clusters de segments

documents[6] dont nous déduisons un clustering *Hard* des documents : chaque document est affecté à sa thématique dominante, c'est à dire au cluster qui lui est le plus proche (en terme de $NCosine$) parmi les clusters contenant au moins l'un de ses segments.

Le tableau 9.4 présente les résultats obtenus avec deux types de clusterings préliminaires de segments, C, N ne considérant qu'un critère de cohésion des groupes et M, N y adjoignant l'optimisation d'un critère de similarité des segments leaders avec la requête[7]. Selon les résultats reportés dans ce tableau, on peut remarquer que la considération des segments thématiques permet aux deux approches d'améliorer leurs performances (sauf peut-être sur le corpus AP où les différences ne semblent pas significatives). Lorsque le contexte de clustering n'est pas pris en compte (approche C, N), la considération des segments thématiques permet l'obtention de groupes mieux centrés sur les différentes thématiques abordées par les documents (tel qu'ob-

6. Le clustering des segments est réalisé ici selon la mesure *NCosine*. Néanmoins, de la même façon que pour les clusterings de segments thématiques réalisés en section 7.2.2, nous fixons arbitrairement la similarité entre deux segments provenant d'un même document, qui puisqu'ayant été probablement rédigés par une même personne ont des chances de présenter une trop forte similarité malgré leurs différences thématiques, de telle sorte qu'elle corresponde à la similarité moyenne entre segments de documents différents.

7. Les deux approches concernent le partitionnement en cinq clusters des segments des 50 premiers documents retournés par une recherche préliminaire. Les documents sont ordonnés dans les clusters finaux selon leur similarité avec leur représentant de groupe, c'est à dire le document le plus proche de la requête.

servé au chapitre 7). Lorsque la proximité des groupes à la requête est prise en compte (approche M, N), l'individualisation des passages thématiques des documents conduit à la production de groupes mieux organisés autour du sujet recherché par l'utilisateur.

9.3.2 Que présenter à l'utilisateur ?

Ainsi, l'individualisation des thématiques des documents peut permettre une distinction plus fine des différents aspects de la requête. Reste que la présentation de groupes de documents entiers n'est peut-être pas le meilleur moyen d'aider un utilisateur à appréhender la structure de l'information qu'il recherche. Plutôt que de déduire des clusters de documents à partir d'un partitionnement préliminaire de leurs segments thématiques, nous expérimentons ici les bénéfices pouvant résulter de l'utilisation directe des groupes de segments pour produire un document composite regroupant les segments les plus représentatifs des principaux aspects des informations que l'utilisateur peut trouver dans le corpus interrogé en rapport avec le besoin qu'il a exprimé.

Tit.	ZIFF		AP		WSJ		FR	
	Doc	*Seg*	*Doc*	*Seg*	*Doc*	*Seg*	*Doc*	*Seg*
NbP	2.21	2.52	1.85	2.02	2.66	2.93	3.48	3.62
PrP	0.66	0.66	0.67	0.66	0.66	0.66	0.75	0.76
PRO	0.45	0.50	0.41	0.43	0.55	0.59	0.56	0.60
LAR	0.48	0.56	0.42	0.43	0.55	0.60	0.60	0.67
OPT	0.74	0.79	0.69	0.71	0.82	0.85	0.83	0.90
PM2	0.54	0.61	0.49	0.52	0.66	0.72	0.69	0.75
PM3	0.58	0.64	0.53	0.55	0.67	0.72	0.73	0.80
Nar.	ZIFF		AP		WSJ		FR	
	Doc	*Seg*	*Doc*	*Seg*	*Doc*	*Seg*	*Doc*	*Seg*
NbP	3.32	3.60	2.88	2.97	3.37	3.54	3.71	3.86
PrP	0.68	0.68	0.68	0.67	0.70	0.70	0.77	0.77
PRO	0.55	0.62	0.54	0.55	0.59	0.64	0.56	0.62
LAR	0.62	0.66	0.59	0.61	0.63	0.68	0.62	0.70
OPT	0.83	0.89	0.83	0.85	0.87	0.92	0.87	0.93
PM2	0.67	0.72	0.65	0.68	0.74	0.80	0.74	0.79
PM3	0.71	0.76	0.68	0.70	0.75	0.82	0.76	0.84

TABLE 9.5 – Clusters de documents Vs. Clusters de segments

Nous ne disposons malheureusement pas de jugements de pertinence particuliers pour les différents segments des documents. Nous supposons alors comme pertinents l'ensemble des segments contenus par un document pertinent et nous utilisons la mesure de précision des termes retournés introduite en section 6.2 pour inclure

la longueur de textes dans l'évaluation : la précision moyenne des termes retournés $MATP$, qui rend compte du volume d'informations pertinentes collectées par rapport au volume d'efforts à fournir pour les localiser, est appliquée aux différentes routes PRO, LAR, OPT [8], $PM2$ et $PM3$ reportées dans le tableau 9.5 pour les clusters de documents déduits des clusters des segments (approche Doc) et aux clusters de segments eux-mêmes (approche Seg) [9]. Les résultats présentés dans cette table montrent que le fait de présenter directement les groupes de segments à l'utilisateur peut lui permettre d'atteindre les informations qu'il recherche plus facilement, le ratio de termes à examiner pour rencontrer des fragments de texte potentiellement pertinents étant significativement plus faible lors de la considération de groupes de segments. Le nombre NbP de représentants pertinents rend lui aussi compte des bénéfices résultant de la présentation de segments thématiques à l'utilisateur. Ici les représentants des clusters sont sélectionnés en choisissant le segment le plus proche de la requête dans chaque groupe mais des résultats similaires (et même légèrement meilleurs) ont été enregistrés en utilisant directement les leaders de l'individu extrait de l'archive comme représentants de groupe. Dans ce cas, l'algorithme proposé dans ce chapitre n'est plus uniquement un algorithme de clustering mais constitue un processus de composition de document, cherchant à identifier le sous ensemble de segments les plus représentatifs de thématiques fortement connectées à la requête de l'utilisateur.

9.4 Conclusion

À la lumière de nouvelles méthodologies d'évaluation, nous avons montré au chapitre précédent que le regroupement des documents pertinents dans un même cluster n'était pas nécessairement le meilleur moyen d'aider l'utilisateur dans sa tâche de localisation des informations pertinentes. Plutôt que de modifier les calculs de similarités entre documents pour espérer ainsi la formation de clusters représentant différents aspects de la requête, nous avons proposé ici d'agir directement sur le processus de clustering en attirant les centres des clusters vers la requête de l'utilisateur. En imposant ainsi aux groupes de documents de s'organiser autour du sujet concerné par la recherche d'information, nous permettons une distinction plus fine entre les différents types de documents fortement connectés au besoin exprimé. Cela revient en quelque sorte à chercher un partitionnement de l'espace des similarités en multiples zones de proximité se rejoignant toutes en ce point central que constitue la requête de l'utilisateur. Plus les documents sont proches de la requête, plus ils ont de chances de correspondre au besoin de l'utilisateur et donc plus leurs différences

8. La route optimale est calculée selon la mesure de précision moyenne originale. La précision moyenne des termes retournés est appliquée sur la route finalement obtenue.

9. Les deux approches utilisent un clustering des documents optimisé selon les deux critères de cohésion et de similarité des leaders à la requête. Elles concernent toutes deux les partitionnements des segments des 50 premiers documents retournés par une recherche préliminaire. Les requêtes utilisées sont les topics 1-50 de *TREC*.

ont de l'importance pour distinguer les différents aspects du sujet concerné par la recherche. Les expérimentations réalisées ont permis de mettre en évidence, d'une part la capacité d'un tel procédé à partitionner effectivement l'information pertinente, et d'autre part l'intérêt que cela pouvait avoir pour la localisation des documents intéressants. Lors de son application au niveau des segments thématiques des documents, l'algorithme établi ici représente un processus de recherche des fragments de texte les plus représentatifs des différents aspects de la requête de l'utilisateur. Chaque individu non-dominé de la population finale de l'algorithme constitue un document composite potentiel, chacun de ses gènes représentant l'un des segments à faire figurer dans le document composite final. Plutôt que d'extraire un document unique de la population, il est envisageable de laisser l'utilisateur se promener sur le front de compromis, lui permettant ainsi de sélectionner le degré de concentration/distribution de l'information pertinente qui lui semble le mieux correspondre à ses besoins. Une étude impliquant des utilisateurs réels est nécessaire pour valider notre approche de manière définitive mais les premiers résultats expérimentaux présentés dans ce chapitre paraissent prometteurs.

Conclusion Générale

Dans le but de réduire l'effort à fournir pour localiser l'information pertinente, de nombreux systèmes de recherche d'information ont proposé des modes de présentation des résultats alternatifs à la simple liste ordonnée de documents. Nombre de ces approches s'appuient sur un clustering des documents retournés par un système de recherche initial pour regrouper les documents aux thématiques similaires et ainsi présenter des catégories de résultats permettant une localisation des documents pertinents facilitée [Hearst and Pedersen, 1996; Tombros *et al.*, 2002]. La mise en évidence des relations entre documents du corpus permet alors de guider l'utilisateur dans sa recherche [Croft, 1980]. Néanmoins, outre le problème de la localisation des documents contenant des informations pertinentes, se pose celui de la recherche de ces informations dans le corps de chacun d'entre eux. Chaque document est en effet susceptible d'aborder un certain nombre de thématiques distinctes et, même si un document est fortement connecté à la requête de l'utilisateur, il est possible que les informations pouvant satisfaire les besoins de la recherche soient contenues dans une partie restreinte de son texte. La considération de documents entiers n'est alors peut-être pas nécessairement le meilleur moyen d'aider un utilisateur dans sa recherche d'information. Face à ces observations, le domaine de la recherche d'information s'est élargi depuis un certain nombre d'années à la mise en place d'applications ne visant plus uniquement à aider l'utilisateur dans sa tâche de localisation des documents pertinents mais cherchant à lui construire une réponse synthétique permettant de satisfaire au mieux ses besoins en information [Chaâr, 2003]. Les systèmes, relativement récents, de résumé multi-documents [Mckeown *et al.*, 1999; Dang, 2005], et plus particulièrement les systèmes de résumé orienté requête [Goldstein *et al.*, 2000; Liu *et al.*, 2006], visent à produire un texte reprenant les informations principales qu'un utilisateur pourra trouver dans un corpus en rapport avec sa requête. Ces systèmes synthétisent les informations contenues dans une collection de documents en s'attachant à deux critères majeurs déterminant de la qualité du document produit :

- Pertinence du contenu : le résumé produit doit proposer une couverture maximale du sujet dans un volume de texte donné. Le degré de redondance de l'information doit alors être limité au maximum pour ne présenter que les aspects principaux de la requête formulée par l'utilisateur ;
- Lisibilité du texte : le développement thématique du discours doit être le plus cohérent possible, respectant un ordre "logique" dans les thématiques abordées et limitant au maximum les problèmes de co-référence ou erreurs grammaticales.

Ces deux critères peuvent représenter deux sous-problèmes bien distincts : alors que la recherche d'un contenu pertinent s'apparente à une recherche et un filtrage des informations principales qu'un utilisateur peut trouver en relation avec sa requête, le

second critère de production d'un texte cohérent implique plutôt des notions propres au domaine du traitement du langage. Le travail de thèse présenté ici s'est concentré sur le premier de ces deux sous-problèmes : nous avons cherché à regrouper, dans une entité que nous appelons *document composite* (en opposition au résumé multi-documents qui implique une reformulation et/ou un arrangement des fragments de texte sélectionnés), un sous-ensemble de parties de textes répondant au mieux aux besoins d'information de l'utilisateur.

La production d'une telle entité nous a, dans un premier temps, amené à réfléchir sur le mode de découpage des documents à adopter pour disposer d'extraits de textes adaptés à notre problème. Puisque l'objectif est de produire un document offrant un aperçu des différents aspects du sujet, nous nous sommes orientés vers une segmentation thématique des textes. L'étude réalisée a été l'occasion de proposer deux nouvelles méthodes de segmentation et deux nouvelles méthodologies d'évaluation des méthodes de segmentation thématique. Par l'établissement d'une nouvelle mesure de similarité entre les textes, nous avons montré que ce type de passage permettait une individualisation efficace des thématiques des documents et que leur prise en compte en recherche d'information avait un intérêt non négligeable, tant pour la production d'une liste ordonnée de résultats que pour la production de clusters représentatifs des thématiques abordées par les documents. Une telle individualisation des thématiques permet de produire des groupes de résultats mieux organisés autour des principaux sujets abordés par les documents considérés.

Dans un second temps, nous nous sommes interrogés sur le meilleur mode de sélection des extraits de textes à inclure dans le document composite final. De la même manière que réalisé dans la plupart des systèmes de résumé multi-documents [Stein *et al.*, 2000; Goldstein *et al.*, 2000; Chaâr, 2003], nous avons opté pour l'utilisation de techniques de clustering afin d'identifier les principales thématiques en rapport avec le besoin de l'utilisateur. Néanmoins, le niveau de diversité des textes considérés (qui est par ailleurs accru lorsque l'on manipule des segments thématiques plutôt que des documents), tend à produire un partitionnement dans lequel la majorité des informations pertinentes sont regroupées dans un même cluster. Par conséquent, la sélection de segments dans chacun des groupes ne permet pas l'obtention d'une liste de passages représentant les différents aspects du sujet recherché par l'utilisateur. Après avoir proposé différentes méthodologies d'évaluation des systèmes de recherche d'information réalisant une catégorisation des résultats, il est apparu que la prise en compte de la requête pouvait permettre d'obtenir des groupes mieux organisés autour du besoin de l'utilisateur. Ainsi, nous avons proposé un algorithme de clustering qui cherche un compromis entre un degré optimum de cohésion des groupes formés et un degré de proximité de ces groupes à la requête. De cette manière, le partitionnement est réalisé en considérant avec plus d'importance les différences thématiques entre segments proches du sujet de la recherche, ce qui permet d'obtenir un ensemble de clusters représentatifs des différents aspects de la requête. La constitution de notre document composite est alors finalement réalisée

en sélectionnant un segment représentatif de chacun des groupes formés.

Afin de valider notre approche de composition de documents de manière définitive, une étude impliquant des utilisateurs réels pourrait s'avérer nécessaire mais les premiers résultats expérimentaux présentés dans ce rapport paraissent prometteurs. Nous pensons que le document composite que nous proposons peut s'avérer être un outil de recherche d'information bien plus efficace que les résumés multi-documents puisqu'il ne risque pas de présenter de problèmes de lisibilité induits par l'utilisation d'un processus de reformulation. Certes les passages qui le composent peuvent ne pas s'enchaîner aussi bien que dans un résumé multi-documents mais les extraits de textes présentés ont de grandes chances d'être plus facilement compréhensibles par un utilisateur (puisqu'ayant été rédigés manuellement). De plus, un tel document donne l'opportunité à l'utilisateur d'aller explorer les clusters desquels les segments présentés ont été extraits (ce qui n'est pas évident dans le cas des résumés multi-documents puisque le processus de reformulation implique une certaine rupture avec les textes d'origine). L'objectif n'est alors pas de contenir nécessairement la totalité des informations susceptibles d'intéresser l'utilisateur mais consiste plutôt à lui proposer une sorte de sommaire lui permettant de s'orienter aisément vers les informations qu'il recherche. La production d'un tel sommaire constitue une tâche qui nous semble bien moins ardue que la création d'un résumé devant synthétiser l'ensemble des informations susceptibles d'intéresser l'utilisateur. Nous pensons alors que l'intégration dans les moteurs de recherche actuels de systèmes tels que celui proposé ici est bien plus facilement envisageable que la mise à disposition de systèmes de résumés multi-documents qui, selon Hovy [Hovy, 2005], requièrent de très nombreux progrès dans divers champs de recherche avant de constituer des outils réellement utilisables par le grand public.

Par ailleurs, un résumé multi-documents peut être directement obtenu à partir de notre document composite en y appliquant un processus de résumé automatique (un état de l'art sur les approches de résumé automatique est donné dans [Hovy, 2005]). Après une telle reformulation/réorganisation des extraits de textes contenus par le document composite, nous envisageons de mesurer les apports de notre approche pour la production d'un résumé multi-document orienté requête par une participation à la principale campagne d'évaluation des systèmes de résumé multi-documents, la conférence *DUC* (*Documents Understanding Conference*), qui allie évaluation du contenu par comparaison du résumé obtenu avec des résumés réalisés manuellement et évaluation de la lisibilité du texte par repérage des erreurs grammaticales, problèmes de co-référence, etc... [Dang, 2005]. Afin de mesurer les bénéfices résultant du recentrage des clusters autour du besoin de l'utilisateur, une comparaison entre les résultats obtenus par les résumés déduits de plusieurs types de documents composites est envisageable, notamment entre celui déduit d'un document composite produit par optimisation d'un unique critère de cohésion des groupes formés (approche C, N) et celui résultant d'un document composite issu d'un processus de clustering y adjoignant un critère de proximité des groupes à la

requête (approche M, N) (voir chapitre 9). Par ailleurs, la sélection d'un unique représentant de chaque groupe formé n'est peut-être pas nécessairement le meilleur choix pour la production d'un résumé cherchant à réaliser une bonne couverture de sujet. D'autres types de sélection des extraits dans les groupes peuvent être envisagés, par exemple en choisissant un nombre plus important de segments dans les groupes les plus proches de la requête. Lors de la sélection de plusieurs segments dans un même groupe, les modes de sélection peuvent varier d'un individu à l'autre (voir section 2.4 pour les différents modes de sélection des représentants d'un groupe). Enfin, l'impact de la segmentation des documents peut elle aussi être évaluée au travers de la campagne d'évaluation *DUC*, en y soumettant des résumés déduits de documents composites regroupant différents types de passages, notamment des passages de longueur fixe (voir chapitre 6) ou des passages issus de différents processus de segmentation thématique (voir chapitre 5).

Outre ces efforts d'évaluation qui sont susceptibles d'occuper une grande part de nos travaux futurs, les perspectives de recherche sont nombreuses. Tout d'abord, l'utilisation de ressources sémantiques propres au domaine pour lequel le document composite est produit est susceptible d'en améliorer la qualité à de nombreux niveaux, tant pour les performances de la recherche initiale, que pour la qualité du découpage thématique ou que pour la sélection des extraits de textes à faire figurer dans le document composite. L'ensemble des calculs de similarité réalisés dans notre système sont exclusivement statistiques, une considération de ressources telles qu'un thésaurus ou des ontologies spécifiques aux corpus de documents interrogé pourrait nous permettre de dépasser les seules co-occurrences de termes qui ne permettent qu'un rapprochement approximatif des concepts abordés par les textes manipulés. L'application d'une analyse sémantique telle que proposée par le modèle *LSI* (voir section 1.3.1) pourrait elle aussi être envisagée. Par ailleurs, le domaine et le type de documents du corpus considéré peuvent constituer des informations à prendre en compte lors de la conception de notre document composite. Le type de besoin exprimé par l'utilisateur peut lui aussi influer sur les orientations à prendre pour la production du document composite, on ne composera par exemple pas nécessairement de la même manière que le besoin exprimé concerne des renseignements sur une notion précise ou bien qu'il représente une demande d'informations générales sur un domaine donné. Une réflexion sur les différents types de composition à réaliser est alors certainement nécessaire afin de proposer un système qui soit capable de s'adapter selon les besoins de l'utilisateur et le type de corpus interrogé.

Notre approche, qui considère la constitution d'un aperçu des différents aspects d'un sujet comme un problème d'optimisation multi-objectifs, peut être facilement étendue par l'ajout de critères à optimiser. Dans l'état actuel, seuls deux critères de représentativité (traduit par la similarité des segments avec leur plus proche leader) et de proximité à la requête des extraits de textes à faire figurer dans le document composite sont considérés mais de nombreux autres critères sont envisageables. Afin de produire le meilleur aperçu des informations qu'un utilisateur peut trouver en

rapport avec un besoin exprimé, des critères de dissimilarité des segments (à quel point les segments choisis appartiennent à des thématiques éloignées), d'unicité de l'information apportée (qui peut se traduire par un degré de séparation entre les termes utilisés par les segments choisis) ou de longueur du document composite peuvent être facilement intégrés à notre approche. Certains systèmes de résumés multi-document s'intéressent à un critère de "fraicheur de l'information" (voir par exemple [Goldstein *et al.*, 2000], les dates de publication des documents desquels sont extraits les segments figurant dans notre document composite peuvent constituer un autre critère à optimiser pour privilégier les informations récentes. La segmentation thématique risquant parfois, malgré les efforts fournis pour en améliorer les performances, de couper les documents dans des zones où des relations de co-référence existent, un critère permettant de privilégier les segments ne paraissant pas souffrir de problèmes de co-référence peut être envisagé pour sélectionner les extraits de textes les plus compréhensibles hors contexte. Par ailleurs, bien que l'ensemble des observations réalisées et les approches proposées concerne la recherche d'information textuelle contenue dans des corpus de documents sans liens explicites (tels que des liens hypertextes par exemple), il est tout à fait possible d'envisager l'intégration d'un critère prenant en compte les propriétés de connectivité du graphe représentatif des relations entre documents (préférant par exemple les documents *autorité*, possédant un grand nombre de liens entrants, ou les documents *hub*, possédant de nombreux liens vers des documents connectés au besoin de l'utilisateur [Picarougne, 2004]). Enfin, une interaction avec l'utilisateur peut être mise en place en permettant à l'utilisateur de sélectionner les segments qui l'intéressent le plus parmi les segments du document composite et en relançant le processus en optimisant un critère prenant en compte les choix de l'utilisateur (pour chercher par exemple à attirer les groupes de segments vers les segments jugés intéressants, ce qui pourrait permettre de distinguer des thématiques qui y sont fortement connectées).

Liste des figures

1 Composition de documents 3

1.1 Fonctionnement général d'un système de recherche d'information. 13
1.2 Correspondance entre informativité et fréquence des termes 16
1.3 Évaluation d'une conjonction et d'une disjonction. 23
1.4 Représentation dans le modèle vectoriel. 24
1.5 Probabilité de pertinence / sélection en fonction de la taille des documents 34
1.6 Inclinaison du facteur de normalisation de la mesure Cosine. 35
1.7 Deux exemples de barres de visualisation du contenu 42

2.1 Clustering de données dans un espace en deux dimensions 49
2.2 Exemple de hiérarchie 54
2.3 Distances entre clusters dans les méthodes *Single Link* et *Complete Link* 62

3.1 Requête n 74 de TREC-1 70
3.2 Précision et rappel selon le nombre de documents considérés. 74
3.3 Courbes Rappel / Précision 75
3.4 Parcours à travers les clusters. 79
3.5 Parcours en profondeur et parcours en largeur. 79

4.1 Optimum global et optima locaux d'une fonction 85
4.2 Croisements en 1 et 2 points 89
4.3 Zones de préférence / dominance dans un espace de solutions à deux objectifs 91
4.4 Fonctionnement général 94
4.5 Valeur d'adaptation dans SPEA 96

5.1 Exemple de chaînes lexicales 107
5.2 Exemple de fenêtre de calcul 107
5.3 Fonctionnement de ClassStruggle 119
5.4 Réglages des paramètres α et β de ClassStruggle 121
5.5 Évolution de la population au cours des générations 126
5.6 Résultats des expérimentations sur le paramètre α de SegGen 127
5.7 Extrémités du texte et WindowDiff 134
5.8 Tendances de WindowDiff selon le nombre de frontières déterminées . . . 136
5.9 Tendances de WindowDiff selon le nombre de frontières de référence . . . 137
5.10 Comparaison de segmentations aux nombres de frontières différents . . . 140

6.1 Tendances des scores de pertinence 158
6.2 Valeurs des coefficients 162

7.1 Similarités entre documents 172

8.1 Deux partitionnements de 18 documents en 5 clusters. 192
8.2 Influence de la répartition des documents dans les clusters 205
8.3 Influence de la répartition des documents pertinents dans les clusters . . 206
8.4 Influence du nombre de clusters produits 207

9.1 Effets des critères optimisés sur les groupes de documents formés. 215
9.2 Influence du nombre de documents considérés. 221
9.3 Influence du nombre de clusters produits 222

Listes des tables

1.1 Table de vérité 23
1.2 Présence du terme vs. Pertinence du document 28

2.1 Applications des méthodes de clustering en recherche d'information . . . 50
2.2 Coefficients de Lance et Williams 63

3.1 Statistiques des collections de documents 72

5.1 Évaluation des méthodes de segmentation sur le corpus AP 131
5.2 Évaluation des méthodes de segmentation sur le corpus ZIFF 132
5.3 Résultats avec les mesures d'évaluation classiques 138
5.4 Résultats avec les mesures normalisées 142
5.5 Résultats du test de Stabilité 146

6.1 Efficacité des mesures de pertinence 153
6.2 Informations collectées *Vs.* Longueur des textes 156
6.3 Coefficients de corrélation 163
6.4 Efficacité de la mesure *Cosine* normalisée par régression statistique . . . 163
6.5 Statistiques des corpus réduits 166
6.6 Résultats des approches selon les trois mesures de pertinence 167

7.1 Impacts de la normalisation sur les similarités entre documents 174
7.2 Statistiques des corpus utilisés 177
7.3 Test des plus proches voisins 178
7.4 Performances des approches : qualité du groupe optimal 180

8.1 Notations des mesures d'évaluation 201
8.2 Corrélation entre variations observées 203
8.3 Statistiques des partitionnements produits 208
8.4 Résultats des systèmes 209

9.1 Statistiques des partitionnements produits 219
9.2 Résultats des systèmes 220
9.3 Estimation du nombre de clusters optimal 223
9.4 Clusterings de documents déduits de clusters de segments 225
9.5 Clusters de documents Vs. Clusters de segments 226

Listes des algorithmes

2.1 Algorithme Single-Pass 57
2.2 Algorithme des Nuées Dynamiques 58
2.3 Algorithme K-Means 60
2.4 Algorithme de clustering hiérarchique ascendant 61
4.1 Pseudo-code d'un algorithme génétique classique 88
4.2 Pseudo-code de l'algorithme SPEA 94
5.1 Algorithme de la méthode de segmentation ClassStruggle 117
5.2 Algorithme du Test de Stabilité 144
8.1 Algorithme d'évaluation par construction de la liste optimale 193

Index

Algorithme génétique, 86
Algorithme SPEA, 93
Average Precision, 75

C99, 128
Catégorisation, 48
Centroïde, 55
Chaîne lexicale, 107
Classification, 25
ClassStruggle, 114
Cluster Hypothesis, 49
Clustering, 25, 48
Clustering hiérarchique, 60
Co-occurence, 106
Coefficient de corrélation, 162
Cohésion, 48, 105
Complete Link, 62
Corpus, 69
Cosine, 31
Croisement, 88

Diversification, 86
Document composite, 3, 183, 230
Document structuré, 12
Dominance, 90
DotPlotting, 128
DUC, 231

E-mesure, 80
Expansion de requête, 38

F-mesure, 76
Feedback, 40
Fenêtre de calcul, 107
Fichier inversé, 18
Fitness, 87
Flat Clustering, 53
Fonction objectif, 84
Front Pareto, 90
Frontière thématique, 105

Group Average, 62

Hard Clustering, 48
Heuristique, 85
Hiatus, 107
Hyperonyme, 106
Hyponyme, 106

Indexation, 13
Inquery, 31
Intensification, 86

K-Means, 59

Leader, 214
Lemmatisation, 14
Lemme, 14
Lien hypertexte, 12
Liste ordonnée, 41

Méthode des moindres carrés, 161
Méthodes approchées, 84
Méthodes exactes, 84
Maximum de Vraisemblance, 30
Medoïde, 59
Meta-heuristique, 85
MK1, 80
MK3, 81
Modèle booléen, 20
Modèle de langue, 30
Modèle LSI, 28
Modèle probabiliste, 26

Modèle vectoriel, 24
Mutation, 89

NCosine, 164
Nearest Neighbour Test, 77
Nuées Dynamiques, 58
NWin, 141

Okapi, 31
Optimisation combinatoire, 84
Optimisation multi-objectifs, 90
Optimum global, 85
Optimum local, 85
Overlap Test, 77

Parcours en largeur, 79
Parcours en profondeur, 79
Parcours moyen, 195
Parcours optimal, 192
Passage Retrieval, 36
Pertinence, 68
Pivoted Cosine, 31
Pk-mesure, 110
Polymorphisme, 15
Précision, 73, 110, 156
Précision moyenne, 75
Problème NP-complet, 85
Problème NP-difficile, 85
Proximité des pertinents, 198

QSSM, 188

Régression statistique, 160
Résumé automatique, 64
Résumé multi-documents, 3, 64, 229
Rappel, 73, 110
Recherche d'information, 1
Recherche Tabou, 85
Recuit Simulé, 85
Relevance Feedback, 25, 191
Représentant de cluster, 64
Requête, 69
Retrieval Status Value, 26
Rhème, 104

RI, 1
Roulette wheel, 88

Sélection, 88
Séparation, 48
SegGen, 115
Segment thématique, 104
Similarité, 31
Single Link, 62
Single-Pass, 57
Sliding Window, 107
Soft Clustering, 48
Stemming, 14
Stop-list, 17
Synonyme, 106
Système Question-Réponse, 12

Terme, 14
Test de Stabilité, 143
Test de Student, 133
TextTiling, 128
TF * IDF, 17
Thésaurus, 39
Thème, 104
TNWin, 141
TREC, 69

Voisinage, 85

Window-based Passage, 151
WindowDiff, 111

Références bibliographiques

[Aarts and Lenstra, 1997] cité page 84
Emile Aarts and Jan K. Lenstra, editors. *Local Search in Combinatorial Optimization*. John Wiley & Sons, Inc., New York, NY, USA, 1997.

[Akaike, 1974] cité page 59
Hirotugu Akaike. A new look at the statistical model identification. *IEEE Transactions on automatic control*, 19(6) :716–723, 1974.

[Al-hawamdeh and Willett, 1989] cité page 104
Suliman Al-hawamdeh and Peter Willett. Paragraph-based nearest neighbour searching in full text documents. *Electronic Publishing — Origination, Dissemination, and Design*, 2(4) :179–192, 1989.

[Alvarez *et al.*, 2004] cité page 30
Carmen Alvarez, Philippe Langlais, and Jian-Yun Nie. Mots composés dans les modèles de langue pour la recherche d'information. In *11e édition de la conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, pages 11–16, apr 2004.

[Amati and Van Rijsbergen, 2002] cité page 19
Gianni Amati and Cornelius J. Van Rijsbergen. Probabilistic models of information retrieval based on measuring divergence from randomness. *ACM Transactions on Information Systems*, 20(4) :357–389, 2002.

[Amba *et al.*, 1996] cité page 51
S. Amba, N. Narasimhamurthi, Kevin C. O'Kane, and Philip M. Turner. Automatic linking of thesauri. In *SIGIR '96 : Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 181–186, New York, NY, USA, 1996. ACM.

[Amini *et al.*, 2000] cité page 106
M. Amini, H. Zaragoza, and P. Gallinari. Learning for Sequence Extraction Tasks. In CID, editor, *Proceedings of 6th Conference on Content-Based Multimedia Information Access (RIAO'2000)*, pages 476–490, Paris, France, 2000.

[Anderberg, 1973] cité page 56
M. R. Anderberg. *Cluster analysis for applications*. Probability and Mathematical Statistics, New York : Academic Press, 1973.

[Anick and Vaithyanathan, 1997] cité page 64
Peter G. Anick and Shivakumar Vaithyanathan. Exploiting clustering and phrases for context-based information retrieval. In *SIGIR '97 : Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 314–323, New York, NY, USA, 1997. ACM.

[Azé *et al.*, 2006] cité page 148
J. Azé, T. Heitz, A. Mela, A.-D. Mezaour, P. Peinl, and M. Roche. Présentation de deft'06 (defi fouille de textes). In *Actes de l'atelier DEFT'06, SDN'06 (Semaine du Document Numérique)*, 2006.

[Baeza-Yates and Ribeiro-Neto, 1999] cité page 12, 17, 20, 23, 26, 76
Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.

[Bai *et al.*, 2005] cité page 39
Jing Bai, Dawei Song, Peter Bruza, Jian-Yun Nie, and Guihong Cao. Query expansion using term relationships in language models for information retrieval. In Otthein Herzog, Hans-Jörg Schek, Norbert Fuhr, Abdur Chowdhury, and Wilfried Teiken, editors, *CIKM*, pages 688–695. ACM, 2005.

[Barichard, 2003] cité page 92, 95
Vincent Barichard. *Approches hybrides pour les problèmes multiobjectifs*. PhD thesis, Université d'Angers, 2003.

[Barry, 1998] cité page 65, 65, 65
Carol L. Barry. Document representations and clues to document relevance. *Journal of the American Society of Information Science*, 49(14) :1293–1303, 1998.

[Bates, 1989] cité page 39
Marcia J. Bates. The design of browsing and berrypicking techniques for the online search interface. *Online Review*, 13(5) :407–424, 1989.

[Bäck *et al.*, 1997] cité page 87
Thomas Bäck, David B. Fogel, and Zbigniew Michalewicz, editors. *Handbook of Evolutionary Computation*. Published in cooperation with the Institute of Physics, Ringbound Edition, 1997.

[Beck, 2006] cité page 62, 76
Nicolas Beck. *Application de Méthodes de Clustering Traditionnelles et Extension au Cadre Multicritère*. Master's thesis, Université Libre de Bruxelles, Faculté des Sciences appliquées, 2006.

[Beeferman *et al.*, 1997] cité page 106, 110, 110
D. Beeferman, A. Berger, and J. Lafferty. Text segmentation using exponential models. In Claire Cardie and Ralph Weischedel, editors, *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 35–46. Association for Computational Linguistics, Somerset, New Jersey, 1997.

[Beeferman *et al.*, 1999] cité page 29
D. Beeferman, A. Berger, and J. Lafferty. Statistical models for text segmentation. *Machine Learning*, 34(1-3) :177–210, 1999.

[Belew and Van Rijsbergen, 2000] cité page 9, 12
Richard K. Belew and C. J. Van Rijsbergen. *Finding out about : a cognitive perspective on search engine technology and the WWW*. Cambridge University Press, New York, NY, USA, 2000.

[Belkin *et al.*, 1997] cité page 39
Nicholas J. Belkin, Robert N. Oddy, and Helene M. Brooks. Ask for information retrieval : part i. : background and theory. In *Readings in information retrieval*, pages 299–304. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[Bellot and El-Bèze, 1999] cité page 5, 78, 78, 189
Patrice Bellot and Marc El-Bèze. Query length, number of classes and routes through clusters : Experiments with a clustering method for information retrieval. In *ICSC '99 : Proceedings of the 5th International Computer Science Conference on Internet Applications*, pages 196–205, London, UK, 1999. Springer-Verlag.

[Bellot, 2000] cité page 54, 56, 113, 114, 133, 186
Patrice Bellot. *Méthodes de classification et de segmentation locales non supervisées pour la recherche documentaire*. Phd thesis, Université d'Avignon, Janvier 2000.

[Berger and Lafferty, 1999] cité page 31
Adam Berger and John Lafferty. Information retrieval as statistical translation. In *SIGIR '99 : Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229, New York, NY, USA, 1999. ACM.

[Bergmark *et al.*, 1977] cité page 54
D. Bergmark, Gerard Salton, and A. Wong. Generation and search of clustered files. Technical report, Cornell University, Ithaca, NY, USA, 1977.

[Berkhin, 2006] cité page 53, 56
Pavel Berkhin. A survey of clustering data mining techniques. In Jacob Kogan, Charles Nicholas, and Marc Teboulle, editors, *Grouping Multidimensional Data : Recent Advances in Clustering*, pages 25–71. Springer, 2006.

[Bestgen and Piérard, 2006] cité page 108, 128
Yves Bestgen and S. Piérard. Comment évaluer les algorithmes de segmentation automatiques ? essai de construction d'un matériel de référence. In *Proceedings of TALN'06*, 2006.

[Bezdek, 1981] cité page 59
James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[Bigi *et al.*, 1998] cité page 106
Brigitte Bigi, Renato De Mori, Marc El-bze, and Thierry Spriet. Detecting topic shifts using a cache memory. In *Proceedings of 5 th International Conference on Spoken Language Processing*, pages 2331–2334, 1998.

[Björck, 1996] cité page 161
Å. Björck. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, 1996.

[Boughanem and Soulé-Dupuy, 1992] cité page 19
Mohand Boughanem and Chantal Soulé-Dupuy. Un modèle connexionniste pour la recherche d'informations. *L' Informatique documentaire*, 47 :13–30, 1992.

[Boughanem *et al.*, 2002] cité page 97
M. Boughanem, C. Chrisment, and L. Tamine. On using genetic algorithms for multimodal relevance optimization in information retrieval. *Journal of the American Society of Information Science Technologies*, 53(11) :934–942, 2002.

[Boughanem *et al.*, 2004] cité page 31
Mohand Boughanem, Wessel Kraaij, and Jian-Yun Nie. Modèles de langue pour la recherche d'information. In majid Ihadjadene, editor, *Les systèmes de recherche d'informations*, pages 163–182. Hermes-Lavoisier, Paris, France, 2004.

[Broglio *et al.*, 1994] cité page 32, 33
John Broglio, James P. Callan, W. Bruce Croft, and Daniel W. Nachbar. Document retrieval and routing using the INQUERY system. In *Text REtrieval Conference*, pages 0–, 1994.

[Brown and Yule, 1983] cité page 104
G. Brown and G. Yule. *Discourse Analysis*. Cambridge University Press, Cambridge, MA, 1983.

[Buckland and Gey, 1999] cité page 76
Michael Buckland and Fredric Gey. The relationship between recall and precision. *Journal of the American Society for Information Science*, 45(1) :12–19, January 1999.

[Buckley *et al.*, 1994] cité page 17, 32
Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic query expansion using SMART : TREC 3. In *Text REtrieval Conference*, pages 69–80, 1994.

[Buckley *et al.*, 2006] cité page 71
Chris Buckley, Darrin Dimmick, Ian Soboroff, and Ellen Voorhees. Bias and the limits of pooling. In *SIGIR '06 : Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 619–620, New York, NY, USA, 2006. ACM.

[Burgin, 1995] cité page 52
Robert Burgin. The retrieval effectiveness of five clustering algorithms as a function of indexing exhaustivity. *J. Am. Soc. Inf. Sci.*, 46(8) :562–572, 1995.

[Buvet *et al.*, 2003] cité page 15
Pierre-André Buvet, Fabienne Moreau, and Max Silberztein. Procédures de désambiguïsation pour les systèmes de recherche d'information. *Revue québécoise de linguistique*, 32(1) :177–197, 2003.

[Caillet *et al.*, 2004] cité page 106
M. Caillet, J.F. Pessiot, M. Amini, and P. Gallinari. Unsupervised learning with term clustering for thematic segmentation of texts. In *the 7th Proceedings of Rercherche d'Information Assistée par Ordinateur (RIAO'04)*, pages 1–11, Avignon, March 2004.

[Callan *et al.*, 1995] cité page 26, 28, 31, 32
James P. Callan, W. Bruce Croft, and John Broglio. TREC and tipster experiments with inquery. *Information Processing and Management*, 31(3) :327–343, 1995.

[Callan, 1994] cité page 5, 104, 150, 150, 151, 151, 151, 151, 168
J.P. Callan. Passage-Level Evidence in Document Retrieval. In Bruce W. Croft and Cornelius J. Van Rijsbergen, editors, *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 302 – 310, Dublin, Ireland, July 1994. Spring-Verlag.

[Chafe, 1979] cité page 104
W.L. Chafe. The flow of thought and the flow of language in discourse and syntax. *Syntax and Semantics Ann Arbor, Mich.*, 12 :159–181, 1979.

[Chang and Hsu, 1997] cité page 188
C. Chang and C. Hsu. Customizable multi-engine search tool with clustering. In *Proceedings of the 6th WWW Conference*. ACM, 1997.

[Chang and Hsu, 2005] cité page 176
Hsi-Cheng Chang and Chiun-Chieh Hsu. Using topic keyword clusters for automatic document clustering. In *ICITA '05 : Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05) Volume 2*, pages 419–424, Washington, DC, USA, 2005. IEEE Computer Society.

[Chaâr *et al.*, 2002] cité page 12
Sana Leila Chaâr, Olivier Ferret, and Christian Fluhr. Filtrage multi-document orienté par un profil utilisateur. In *5ième Colloque international sur le document électronique (CIDE'02)*, pages 247–260, Le Chesnay, France, 2002. INRIA.

[Chaâr, 2003] cité page 229, 230
Sana Leila Chaâr. Extraction de segments thématiques pour la construction de résumé multidocument orienté par un profil utilisateur. In *7ième Rencontre des Etudiants-Chercheurs en Informatique et en Traitement Automatique des Langues (RECITAL'03)*, 2003.

[Charniak, 1997] cité page 29
Eugene Charniak. Statistical techniques for natural language parsing. *AI Magazine*, 18 :33–44, 1997.

[Chen and Dumais, 2000] cité page 43, 43
Hao Chen and Susan Dumais. Bringing order to the web : automatically categorizing search results. In *CHI '00 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 145–152, New York, NY, USA, 2000. ACM.

[Chimera and Shneiderman, 1994] cité page 43, 43
Richard Chimera and Ben Shneiderman. An exploratory evaluation of three interfaces for browsing large hierarchical tables of contents. *ACM Transactions on Information Systems*, 12(4) :383–406, 1994.

[Cho *et al.*, 2003] cité page 26
Bong-Hyun Cho, Changki Lee, and Gary Geunbae Lee. Exploring term dependences in probabilistic information retrieval model. *Inf. Process. Manage.*, 39(4) :505–519, 2003.

[Choi, 2000] cité page 109, 128
F.Y.Y. Choi. Advances in domain independent linear text segmentation. In *Proceedings of the first conference on North American chapter of the Association for Computational Linguistics*, pages 26–33, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[Choquette, 1996] cité page 150
Martin Choquette. *Passage Retrieval*. PhD thesis, Fitzwilliam College, Cambridge, 1996.

[Chung *et al.*, 2006] cité page 35, 157, 158, 158
Tze Leung Chung, Robert Wing Pong Luk, Kam Fai Wong, Kui Lam Kwok, and Dik Lun Lee. Adapting pivoted document-length normalization for query size : Experiments in chinese and english. *ACM Transactions on Asian Language Information Processing (TALIP)*, 5(3) :245–263, 2006.

[Collette and Siarry, 2002] cité page 92
Y. Collette and P. Siarry. *Optimisation multiobjectif*. Eyrolles, 2002.

[Conrad *et al.*, 2005] cité page 176
Jack G. Conrad, Khalid Al-Kofahi, Ying Zhao, and George Karypis. Effective document clustering for large heterogeneous law firm collections. In *ICAIL '05 : Proceedings of the 10th international conference on Artificial intelligence and law*, pages 177–187, New York, NY, USA, 2005. ACM Press.

[Cooper *et al.*, 1992] cité page 159
William S. Cooper, Fredric C. Gey, and Aitao Chen. Probabilistic retrieval in the tipster collections : An application of staged logistic regression. In *TREC*, pages 73–88, 1992.

[Cormack *et al.*, 1997] cité page 151
G. Cormack, C. Clarke, C. Palmer, and S. To. Passage-based reffinement (multitext experiments for trec-6). In *Proceedings of the 6th Text REtrieval Conference (TREC-6), NIST Special Publication 500-240*, pages 303–320, 1997.

[Croft and Harper, 1997] cité page 40
W. B. Croft and D. J. Harper. Using probabilistic models of document retrieval without relevance information. In *Readings in information retrieval*, pages 339–344. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[Croft, 1978] cité page 55
W. B. Croft. *Organizing and searching large files of document descriptions*. PhD thesis, Churchill College, University of Cambridge, 1978.

[Croft, 1980] cité page 1, 229
W. Bruce Croft. A model of cluster searching bases on classification. *Information Systems*, 5(3) :189–195, 1980.

[Crouch and Yang, 1992] cité page 51
Carolyn J. Crouch and Bokyung Yang. Experiments in automatic statistical thesaurus construction. In *SIGIR '92 : Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 77–88, New York, NY, USA, 1992. ACM.

[Cucerzan and Brill, 2004] cité page 39
Silviu Cucerzan and Eric Brill. Spelling correction as an iterative process that exploits the collective knowledge of web users. In *Empirical Methods in Natural Language Processing*, 2004.

[Cugini *et al.*, 2000] cité page 44
John Cugini, S. Laskowski, and M. Sebrechts. Design of 3d visualization of search results : Evolution and evaluation. In *Proceedings of IST/SPIE's 12th Annual International Symposium : Electronic Imaging 2000 : Visual Data Exploration and Analysis (SPIE 2000)*, pages 23–28, 2000.

[Cutting *et al.*, 1992] cité page 43, 186
Douglass R. Cutting, Jan O. Pedersen, David Karger, and John W. Tukey. Scatter/gather : A cluster-based approach to browsing large document collections. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 318–329, 1992.

[Dang, 2005] cité page 3, 64, 229, 231
Hoa Trang Dang. Overview of DUC 2005. In *Document Understanding Conferences (DUC'05)*, Rochester, New York, USA, 2005. Hyatt Regency Rochester.

[Darwin, 1859] cité page 86
Charles Darwin. *The Origin of Species by Means of Natural Selection*. Mentor Reprints, 1958, 1859.

[De Jong, 1975] cité page 93
K.A. De Jong. *Analysis of Behavior of a Class of Genetic Adaptive Systems*. Phd thesis, University of Michigan, 1975.

[de Loupy and Bellot, 2000] cité page 71
C. de Loupy and P. Bellot. Evaluation of document retrieval systems and query

difficulty. In *LREC'2000 Satelitte Workshop : "Using Evaluation within HLT Programs"*, may 2000.

[de Loupy *et al.*, 1999] cité page 186
C. de Loupy, P. Bellot, M. El-Bèze, and P.-F. Marteau. Query expansion and automatic classification. In *Text REtrieval Conference TREC-7, NIST special publication 500-242*, pages 443–450, 1999.

[de Loupy, 2000] cité page 71
Claude de Loupy. *Évaluation de l'Apport de Connaissances Linguistiques en Désambiguïsation Sémantique et Recherche Documentaire*. Phd thesis, Université d'Avignon et des Pays de Vaucluse, Novembre 2000.

[Deerwester *et al.*, 1990] cité page 28
Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6) :391–407, 1990.

[Dempster *et al.*, 1977] cité page 56
A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B*, 39 :1–38, 1977.

[Diday *et al.*, 1982] cité page 55, 56, 58, 58
Edwin Diday, Jacques Lemaire, Jean Pouget, and Francoise Testu. *Eléments d'analyse de données*. Dunod Informatique, Paris, France, 1982.

[Dinet, 2000] cité page 37
Jérôme Dinet. La pertinence des outils d'experts au service des non-experts en recherche d'informations : un exemple avec les opérateurs booléens. *Revue de l'EPI*, 99 :57–68, 2000.

[Dorigo and Caro, 1999] cité page 84
Marco Dorigo and Gianni Di Caro. The ant colony optimization meta-heuristic. In *New ideas in optimization*, pages 11–32. McGraw-Hill Ltd., UK, Maidenhead, UK, England, 1999.

[Doyle, 1964] cité page 51
Laurent B. Doyle. Some compromises between word grouping and document grouping. In *Proceedings of the Symposium on Statistical Association Methods for Mechanized Documentation*, pages 15–24. U.S. Department of Commerce, national Bureau of Standards, Misc. Publication 269, 1964.

[Drori and Alon, 2003] cité page 43, 43
Offer Drori and Nir Alon. Using documents classification for displaying search results list. *Journal of Information Science*, 29 :97–106, 2003.

[Drori, 2000] cité page 42, 42
Offer Drori. The benefits of displaying additional internal document information on textual database search result lists. In *ECDL '00 : Proceedings of the 4th*

European Conference on Research and Advanced Technology for Digital Libraries, pages 69–82, London, UK, 2000. Springer-Verlag.

[Du and Pardalos, 1998] cité page 84
Ding-Zhu Du and P.M. Pardalos, editors. *Handbook of Combinatorial Optimization*, volume 1-3. Springer, 1998.

[Dumais *et al.*, 1998] cité page 48
Susan Dumais, John Platt, David Heckerman, and Mehran Sahami. Inductive learning algorithms and representations for text categorization. In *CIKM '98 : Proceedings of the seventh international conference on Information and knowledge management*, pages 148–155, New York, NY, USA, 1998. ACM.

[Dumais, 1994] cité page 29
Susan T. Dumais. Latent semantic indexing (lsi) and trec-2. In *The Second Text REtrieval Conference (TREC2), National Institute of Standards and Technology Special Publication 500-215*, pages 105–116, 1994.

[Ehrgott and Gandibleux, 2000] cité page 92
M. Ehrgott and X. Gandibleux. A Survey and Annotated Bibliography of Multiobjective Combinatorial Optimization. *OR Spektrum*, 22 :425–460, 2000.

[El-Bèze and Spriet, 1995] cité page 15
Marc El-Bèze and Thierry Spriet. Intégration de contraintes syntaxiques dans un systèmes d'étiquetage probabiliste. *Traitement automatique des langues*, 36(1-2) :47–66, 1995.

[Ellis *et al.*, 1993] cité page 52
D. Ellis, J. Furner-Hines, and P. Willett. Measuring the degree of similarity between objects in text retrieval systems. *Perspectives in Information Management*, 3(2) :128–149, 1993.

[Ferret, 2002] cité page 108
Olivier Ferret. Segmenter et structurer thématiquement des textes par l'utilisation conjointe de collocations et de la récurrence lexicale. In *Proceedings of TALN'02*, pages 155–165, 2002.

[Fisher *et al.*, 2003] cité page 98
Michelle J. Fisher, Jonathan E. Fieldsend, and Richard M. Everson. Multi-objective optimisation for information access tasks draft submitted to cikm 2003 abstract, 2003.

[Fogel *et al.*, 1966] cité page 87
L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley, New York, USA, 1966.

[Fonseca and Fleming, 1995] cité page 86
Carlos M. Fonseca and Peter J. Fleming. An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation*, 3 :1–16, 1995.

[Fox, 1983] cité page 23
Edward Alan Fox. *Extending the boolean and vector space models of information retrieval with p-norm queries and multiple concept types*. PhD thesis, Cornell University, Ithaca, NY, USA, 1983.

[Fox, 1992] cité page 17, 45
Chris Fox. Lexical analysis and stop lists. In Frakes and Baeza-Yates [1992], pages 102–130.

[Frakes and Baeza-Yates, 1992] cité page 18, 18, 57, 252
William B. Frakes and Ricardo A. Baeza-Yates, editors. *Information Retrieval : Data Structures & Algorithms*. Prentice-Hall, 1992.

[Friedman and Goldszmidt, 1996] cité page 26
Nir Friedman and Moises Goldszmidt. Building classifiers using Bayesian networks. In *Proc. National Conference on Artificial Intelligence*, pages 1277–1284, 1996.

[Fuhr *et al.*, 2006] cité page 12
Norbert Fuhr, Mounia Lalmas, Saadia Malik, and Gabriella Kazai. *Advances in XML Information Retrieval and Evaluation : 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl ... Papers (Lecture Notes in Computer Science)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[Fuhr, 1989] cité page 26
Norbert Fuhr. Models for retrieval with probabilistic indexing. *Inf. Process. Manage.*, 25(1) :55–72, 1989.

[Galley *et al.*, 2003] cité page 106
M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing. Discourse segmentation of multi-party conversation. In *ACL '03 : Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 562–569, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

[Garofolo *et al.*, 1999] cité page 69
John S. Garofolo, Ellen M. Voorhees, Cedric G. P. Auzanne, Vincent M. Stanford, and Bruce A. Lund. 1998 trec-7 spoken document retrieval track overview and results. In *Proceedings of the 7th Text Retrieval Conference TREC-7*, 1999.

[Gelbukh *et al.*, 2003] cité page 56, 64
Alexander F. Gelbukh, Mikhail Alexandrov, Ales Bourek, and Pavel Makagonov. Selection of representative documents for clusters in a document collection. In Antje Düsterhöft and Bernhard Thalheim, editors, *NLDB*, volume 29 of *LNI*, pages 120–126. GI, 2003.

[Glover and Kochenberger, 2003] cité page 84
F. Glover and G. Kochenberger, editors. *Handbook of Metaheuristics*. Kluwer Academic Publishers, 2003.

[Glover, 1986] cité page 84, 84
Fred Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5) :533–549, 1986.

[Goffman, 1968] cité page 68
William Goffman. An indirect method of information retrieval. *Information Storage and Retrieval*, 4(4) :361–373, 1968.

[Goldberg, 1989] cité page 87, 124, 217
D.E. Goldberg. *Genetic Algorithms in Search Optimization and Machine Learning*. Addison-Wesley, Reading, MA, USA, 1989.

[Goldstein *et al.*, 2000] cité page 3, 229, 230, 232
Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. Multi-document summarization by sentence extraction. In *NAACL-ANLP 2000 Workshop on Automatic summarization*, pages 40–48, Morristown, NJ, USA, 2000. Association for Computational Linguistics.

[Good, 1958] cité page 49
I. J. Good. Speculations concerning information retrieval. Technical report, Research report PC-78, IBM Research Centre, New York, USA, 1958.

[Goodman, 2001] cité page 30
Joshua T. Goodman. A bit of progress in language modeling. *Computer speech & language*, 15(4) :403–434, 2001.

[Google, 2008] cité page 18, 37, 41, 49
Google, 2008.

[Gordon, 1987] cité page 48
M. Gordon. A review of hierarchical classification. *Journal of the Royal Statistical Society. Series A (General)*, 150(2) :119–137, 1987.

[Gordon, 1988] cité page 97
M. Gordon. Probabilistic and genetic algorithms in document retrieval. *Communications of the ACM*, 31(10) :1208–1218, 1988.

[Griffiths *et al.*, 1997] cité page 63
Alan Griffiths, H. Clair Luckhurst, and Peter Willett. Using interdocument similarity information in document retrieval systems. In *Readings in information retrieval*, pages 365–373. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[Grosz and Sidner, 1986] cité page 105, 106
Barbara J. Grosz and Candace L. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3) :175–204, 1986.

[Halliday and Hasan, 1976] cité page 106
M. A. K. Halliday and Ruqaiya Hasan. *Cohesion in English (English Language)*. Longman Pub Group, May 1976.

[Handl and Knowles, 2007] cité page 98, 217, 222
J. Handl and J. Knowles. An evolutionary approach to multiobjective clustering. *Evolutionary Computation, IEEE Transactions on*, 11(1) :56–76, 2007.

[Harman, 1992a] cité page 68
Donna Harman. Evaluation issues in information retrieval. *Information Processing & Management*, 28(4) :439–440, 1992.

[Harman, 1992b] cité page 17
Donna Harman. Ranking algorithms. In Frakes and Baeza-Yates [1992], pages 363–392.

[Harman, 1994] cité page 70
Donna Harman. Overview of the third text retrieval conference (trec-3). In *TREC*, 1994.

[Harman, 1995] cité page 69
Donna Harman. The trec conferences. In *HIM*, pages 9–28, 1995.

[Hatzivassiloglou *et al.*, 2000] cité page 52
Vasileios Hatzivassiloglou, Luis Gravano, and Ankineedu Maganti. An investigation of linguistic features and clustering algorithms for topical document clustering. In *SIGIR 2000*, pages 224–231, 2000.

[Hawking *et al.*, 1999] cité page 69
David Hawking, Nick Craswell, and Paul Thistlewaite. Overview of trec-7 very large collection track. In *Proceedings of the 7th Text Retrieval Conference TREC-7*, pages 91–104, 1999.

[Hearst and Karadi, 1997] cité page 41
Marti A. Hearst and Chandu Karadi. Cat-a-cone : an interactive interface for specifying searches and viewing retrieval results using a large category hierarchy. *SIGIR Forum*, 31(SI) :246–255, 1997.

[Hearst and Pedersen, 1996] cité page 1, 78, 186, 189, 229
Marti A. Hearst and Jan O. Pedersen. Reexamining the cluster hypothesis : Scatter/gather on retrieval results. In *Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval*, pages 76–84, Zürich, CH, 1996.

[Hearst and Plaunt, 1993] cité page 150, 151
M. A. Hearst and C. Plaunt. Subtopic structuring for full-length document access. In *Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 56–68. Association for Computing Machinery, 1993.

[Hearst, 1994] cité page 42, 105, 106, 107, 118, 122, 128
Marti A. Hearst. Context and structure in automated full-text information access. Technical report, University of California at Berkeley, Berkeley, CA, USA, 1994.

[Hearst, 1995] cité page 42
Marti A. Hearst. Tilebars : visualization of term distribution information in full text information access. In *CHI '95 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 59–66, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[Hearst, 1997] cité page 104, 104, 105, 108, 108
M.A. Hearst. Texttiling : segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1) :33–64, 1997.

[Heyer *et al.*, 1999] cité page 59
L. J. Heyer, S. Kruglyak, and S. Yooseph. Exploring expression data : Identification and analysis of coexpressed genes. *Genome Research*, 9(11) :1106–1115, 1999.

[Hiemstra, 2001] cité page 31
Djoerd Hiemstra. *Using language models for information retrieval*. PhD thesis, Centre for Telematics and Information Technology, University of Twente, 2001.

[Hofmann, 1999] cité page 29
Thomas Hofmann. Probabilistic Latent Semantic Indexing. In *Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*, pages 50–57, Berkeley, California, August 1999.

[Holland, 1975] cité page 86, 87, 87, 88
J.H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI, USA, 1975.

[Hong *et al.*, 2007] cité page 23
Won-Sin Hong, Shi-Jay Chen, Li-Hui Wang, and Shyi-Ming Chen. A new approach for fuzzy information retrieval based on weighted power-mean averaging operators. *Comput. Math. Appl.*, 53(12) :1800–1819, 2007.

[Horng and Yeh, 2000] cité page 97
Jorng-Tzong Horng and Ching-Chang Yeh. Applying genetic algorithms to query optimization in document retrieval. *Information Processing & Management*, 36(5) :737–759, 2000.

[Hovy, 2005] cité page 231, 231
Eduard Hovy. Automated text summarization. In *The oxford handbook of computational linguistics*, pages 583–598. Oxford University Press, Oxford, 2005.

[Iwayama, 2000] cité page 6, 188
M. Iwayama. Relevance feedback with a small number of relevance judgements : incremental relevance feedback vs. document clustering. In *Proceedings of the 23rd Annual ACM SIGIR Conference*, pages 10–16. ACM, 2000.

[Jacobs, 1992] cité page 154
P. S. Jacobs. Introduction : Text power and intelligent systems. In P. S. Jacobs, editor, *Text-Based Intelligent Systems : Current Research and Practice in Information Extraction and Retrieval*, pages 1–8. Erlbaum, Hillsdale, 1992.

[Jacques and Rebeyrolle, 2006] cité page 104
Marie-Paule Jacques and Josette Rebeyrolle. Titres et structuration des documents. In *Colloque International Discours et Document*, pages 1–12, Caen, France, 2006. Presses universitaires de Caen.

[Jardine and Van Rijsbergen, 1971] cité page 5, 49, 55, 77, 80, 80, 81, 81, 81, 189
N. Jardine and Cornelis Joost Van Rijsbergen. The use of hierarchic clustering in information retrieval. *Information Storage and Retrieval*, 7(5) :217–240, 1971.

[Jelinek, 1997] cité page 29
Frederick Jelinek. *Statistical methods for speech recognition*. MIT Press, Cambridge, MA, USA, 1997.

[Ji and Zha, 2003] cité page 114, 129, 129, 147
X. Ji and H. Zha. Domain-independent text segmentation using anisotropic diffusion and dynamic programming. In *Proceedings of the 26 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 322–329, 2003.

[Jones, 1971] cité page 51
Karen Sparck Jones. *Automatic keyword classification for information retrieval*. Butterworths, London, UK, UK, 1971.

[Jones, 1988] cité page 17
Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. In *Document retrieval systems*, pages 132–142. Taylor Graham Publishing, London, UK, UK, 1988.

[Jong and Spears, 1993] cité page 84, 86
Kenneth A. De Jong and William M. Spears. On the state of evolutionary computation. In *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 618–625, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.

[Jong, 1992] cité page 86
Kenneth A. De Jong. Are genetic algorithms function optimizers ? In Reinhard Männer and Bernard Manderick, editors, *PPSN*, pages 3–14. Elsevier, 1992.

[Jung *et al.*, 2007] cité page 40
Seikyung Jung, Jonathan L. Herlocker, and Janet Webster. Click data as implicit relevance feedback in web search. *Inf. Process. Manage.*, 43(3) :791–807, 2007.

[Kaszkiel and Zobel, 1997] cité page 151
Marcin Kaszkiel and Justin Zobel. Passage retrieval revisited. In *SIGIR '97 : Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 178–185, New York, NY, USA, 1997. ACM Press.

[Kaszkiel and Zobel, 2001] cité page 5, 36, 150, 151, 151, 151, 164, 165, 165, 168, 168
Marcin Kaszkiel and Justin Zobel. Effective ranking with arbitrary passages. *Journal of the American Society of Information Science*, 52(4) :344–364, 2001.

[Kaszkiel *et al.*, 1999] cité page 18
Marcin Kaszkiel, Justin Zobel, and Ron Sacks-Davis. Efficient passage ranking for document databases. *ACM Transactions on Informaton Systems*, 17(4) :406–439, 1999.

[Kaufman and Rousseeuw, 1990] cité page 59
Leonard Kaufman and Peter J. Rousseeuw. *Finding groups in data*. Wiley, New York, 1990.

[Kaufman and Rousseeuw, 2005] cité page 59, 59
Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data : An Introduction to Cluster Analysis (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2005.

[Kehagias *et al.*, 2003] cité page 114
Athanasios Kehagias, Pavlina Fragkou, and Vassilios Petridis. Linear text segmentation using a dynamic programming algorithm. In *EACL*, pages 171–178, 2003.

[Kent *et al.*, 1955] cité page 81
Allen Kent, Madeline M. Berry, Luehrs, and J. W. Perry. Machine literature searching viii, operational criteria for designing information retrieval systems. *American Documentation*, 6(2) :93–101, 1955.

[Kirkpatrick *et al.*, 1983] cité page 84
S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598) :671–680, May 1983.

[Kishida, 2005] cité page 75
Kazuaki Kishida. Property of average precision and its generalization : an examination of evaluation indicator. Technical Report NII-2005-014E, NII Technical Reports, 2005.

[Kleinberg, 1999] cité page 36
Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5) :604–632, 1999.

[Koshman *et al.*, 2006] cité page 50
Sherry Koshman, Amanda Spink, and Bernard J. Jansen. Web searching on the vivisimo search engine. *Journal of the American Society for Information Science and Technology*, 57(14), DEC 2006.

[Kowalski, 1997] cité page 24
Gerald Kowalski. *Information Retrieval Systems : Theory and Implementation*. Kluwer Academic Publishers, Norwell, MA, USA, 1997.

[Kozima, 1993] cité page 104
H. Kozima. Text segmentation based on similarity between words. In *Meeting of the Association for Computational Linguistics*, pages 286–288, 1993.

[Krovetz and Croft, 1992] cité page 15
Robert Krovetz and W. Bruce Croft. Lexical ambiguity and information retrieval. *ACM Trans. Inf. Syst.*, 10(2) :115–141, 1992.

[Kugel, 1962] cité page 18
Peter Kugel. Information retrieval ii : A data structure for data retrieval. In *Proceedings of the 1962 ACM national conference on Digest of technical papers*, page 110, New York, NY, USA, 1962. ACM. Moderator-A. Kent.

[Kural, 1999] cité page 65
Y. Kural. *Clustering information retrieval search outputs*. PhD thesis, City University, London, UK, 1999.

[Kwok *et al.*, 2001] cité page 37
Cody Kwok, Oren Etzioni, and Daniel S. Weld. Scaling question answering to the web. *ACM Trans. Inf. Syst.*, 19(3) :242–262, 2001.

[Kwok, 1996] cité page 17
Kui-Lam Kwok. A new method of weighting query terms for ad-hoc retrieval. In *SIGIR '96 : Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 187–195, New York, NY, USA, 1996. ACM.

[Lafferty and Zhai, 2001] cité page 31
John Lafferty and Chengxiang Zhai. Document language models, query models, and risk minimization for information retrieval. In *Research and Development in Information Retrieval*, pages 111–119, 2001.

[Lallich-Boivin *et al.*, 2006] cité page 37
Geneviève Lallich-Boivin, Dominique Maret, and Serge Chambaud, editors. *Recherche d'information et traitement de la langue : Fondements linguistiques et applications*. ENSSIB, 2006.

[Lamping *et al.*, 1995] cité page 44
John Lamping, Ramana Rao, and Peter Pirolli. A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proceedings of the ACM Conference on Human Factors in Computing Systems, CHI*, pages 401–408. ACM, 1995.

[Lamprier *et al.*, 2007a] cité page 4
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Classstruggle : a clustering based text segmentation method. In *SAC'07 : The 22nd Annual ACM Symposium on Applied Computing*, pages 600–604, New York, NY, USA, 2007. ACM Press.

[Lamprier *et al.*, 2007b] cité page 4, 5
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Document length normalization by statistical regression. In *19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007)*, volume 2, pages 11–18. IEEE Computer Society, 2007.

[Lamprier *et al.*, 2007c] cité page 4
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. On evaluation methodologies for text segmentation algorithms. In *19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007)*, volume 2, pages 19–26. IEEE Computer Society, 2007.

[Lamprier *et al.*, 2007d] cité page 4
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Optimisation multiobjectif pour la segmentation thématique de textes. In *Conférence scientifique conjointe en Recherche Opérationnelle et Aide à la Décision (Francoro / ROADEF'07)*, pages 279–280, 2007.

[Lamprier *et al.*, 2007e] cité page 4
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Seggen : A genetic algorithm for linear text segmentation. In Manuela M. Veloso, editor, *IJCAI'07*, pages 1647–1652, 2007.

[Lamprier *et al.*, 2008a] cité page 5
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Classification en recherche d'information : Utilisation de segments thématiques. In *5ième Colloque sur l'Optimisation et les Systèmes d'Information (COSI'08)*, pages 206–217, 2008.

[Lamprier *et al.*, 2008b] cité page 6, 6
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Composite documents conception : Multi-objective optimisation for information retrieval. *Knowledge and Information Systems*, 2008. En soumission.

[Lamprier *et al.*, 2008c] cité page 4
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Segmentation thématique : Unité du texte vs indépendance des segments. In *16ième congrès francophone AFRIF-AFIA Reconnaissance des Formes et Intelligence Artificielle (RFIA'08)*, pages 16–23, 2008.

[Lamprier *et al.*, 2008d] cité page 5
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Segmenting texts : a way to improve the cluster based information retrieval. *Information Processing Letters*, 2008. En soumission.

[Lamprier *et al.*, 2008e] cité page 5
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Thematic segment retrieval revisited. In *Artificial Intelligence : Methodology, Systems, and Applications, 13th International Conference, AIMSA'08*, volume 5253 of *Lecture Notes in Computer Science*, pages 157–166, 2008.

[Lamprier *et al.*, 2008f] cité page 4
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Toward a more global and coherent segmentation of texts. *Applied Artificial Intelligence*, 22(3) :208–234, 2008.

[Lamprier *et al.*, 2008g] cité page 4
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Using an evolving thematic clustering in a text segmentation process. *Journal of Universal Computer Science*, 14(2) :178–192, 2008.

[Lamprier *et al.*, 2008h] cité page 5
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Using text segmentation to enhance the cluster hypothesis. In *Artificial Intelligence : Methodology, Systems, and Applications, 13th International Conference, AIMSA'08*, volume 5253 of *Lecture Notes in Computer Science*, pages 69–82, 2008.

[Lamprier *et al.*, 2009] cité page 5, 6
Sylvain Lamprier, Tassadit Amghar, Bernard Levrat, and Frédéric Saubion. Traveling among clusters : a way to reconsider the benefits of the cluster hypothesis. In *31st European Conference on Information Retrieval (ECIR'09)*, 2009. En soumission.

[Lancaster, 1968] cité page 12
F.W. Lancaster. *Information retrieval systems : characteristics, testing, and evaluation.* Wiley, 1968.

[Lance and Williams, 1967] cité page 62
G. N. Lance and W. T. Williams. A general theory of classificatory sorting strategies 1. Hierarchical systems. *Computer Journal*, 9(4) :373–380, February 1967.

[Landi *et al.*, 1998] cité page 69
Bruno Landi, Patrick Kremer, and Laurent Schmitt. Amaryllis : an evaluation experiment on search engine in a french-speaking context. In *Proceedings of the First International Conference on Language Resources & Evaluation (LREC)*, pages 1211–1214, 1998.

[Language, 1991] cité page 107
Language. A new tool for discourse analysis : the vocabulary management profile. *Language*, 67 :763–789, 1991.

[Lebart *et al.*, 2000] cité page 54
Ludovic Lebart, Alain Morineau, and Marie Piron. *Statistique exploratoire multidimensionnelle.* Dunod Informatique, Paris, France, 2000.

[Leuski, 2001a] cité page 50, 64, 191, 191, 197, 197, 197, 201, 205, 208, 208
Anton Leuski. Evaluating document clustering for interactive information retrieval. In *CIKM '01 : Proceedings of the tenth international conference on Information and knowledge management*, pages 33–40, New York, NY, USA, 2001. ACM.

[Leuski, 2001b] cité page 1, 5, 18, 65
Anton V. Leuski. *Interactive information organization : techniques and evaluation.* PhD thesis, University of Amhert, Massachussets, 2001. Director-James Allan.

[Lewis and Jones, 1996] cité page 13
David D. Lewis and Karen Sparck Jones. Natural language processing for information retrieval. *Communications of the ACM*, 39(1) :92–101, 1996.

[Lewis, 1992] cité page 64
David D. Lewis. An evaluation of phrasal and clustered representations on a text categorization task. In Nicholas J. Belkin, Peter Ingwersen, and Annelise Mark Pejtersen, editors, *SIGIR*, pages 37–50. ACM, 1992.

[Liljenback, 2007] cité page 12
Martin Erik Liljenback. *ContextQA : Experiments in Interactive Restricted-Domain Question Answering*. Master's thesis, Faculty of San Diego State University, 2007.

[Liu and Croft, 2004] cité page 49, 51
Xiaoyong Liu and W. Bruce Croft. Cluster-based retrieval using language models. In *SIGIR '04 : Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 186–193, New York, NY, USA, 2004. ACM.

[Liu *et al.*, 2006] cité page 3, 98, 229
Dexi Liu, Yanxiang He, Donghong Ji, and Hua Yang. Genetic algorithm based multi-document summarization. In Qiang Yang and Geoffrey I. Webb, editors, *PRICAI*, volume 4099 of *Lecture Notes in Computer Science*, pages 1140–1144. Springer, 2006.

[Lobo *et al.*, 2007] cité page 89, 125
Fernando G. Lobo, Cláudio F. Lima, and Zbigniew Michalewicz, editors. *Parameter Setting in Evolutionary Algorithms*, volume 54 of *Studies in Computational Intelligence*. Springer, 2007.

[López-Pujalte *et al.*, 2003] cité page 97
Cristina López-Pujalte, Vicente P. Guerrero-Bote, and Félix de Moya-Anegón. Genetic algorithms in relevance feedback : a second test and new contributions. *Information Processing & Management*, 39(5) :669–687, 2003.

[Losee, 2001] cité page 15
Robert M. Losee. Term dependence : a basis for luhn and zipf models. *Journal of the American Society of Information Science*, 52(12) :1019–1025, 2001.

[Luhn, 1958] cité page 13, 16
Hans Peter Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2), 1958.

[Lukasiewicz, 1963] cité page 22
Jan Lukasiewicz. *Elements of Mathematical Logic*. Elsevier, 1963.

[Maarek *et al.*, 2000] cité page 64
Y. S. Maarek, R. Fagin, I. Z. Ben-Shaul, and D. Pelleg. Ephemeral document clustering for web applications. Technical Report RJ 10186, IBM, 2000.

[Malioutov and Barzilay, 2006] cité page 114
Igor Malioutov and Regina Barzilay. Minimum cut model for spoken lecture segmentation. In *ACL '06 : Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, pages 25–32, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[Mani, 2001] cité page 12, 64
Inderjeet Mani. Summarization evaluation : An overview. In *The Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL'01)*, 2001.

[Mann and Thompson, 1988] cité page 105
W. C. Mann and S.A. Thompson. Rhetorical structure theory : Toward a functional theory of text organization. *Text : An Interdisciplinary Journal for the Study of Text*, 8(2) :243–281, 1988.

[Mann, 1999] cité page 41
Thomas M. Mann. Visualization of www-search results. In *DEXA '99 : Proceedings of the 10th International Workshop on Database & Expert Systems Applications*, page 264, Washington, DC, USA, 1999. IEEE Computer Society.

[Manning and Schütze, 1999] cité page 29
Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, Massachusetts, USA, 1999.

[Manning *et al.*, 2008] cité page 12, 25, 26, 26, 38, 39, 48, 49, 49, 59
Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, July 2008.

[Marchionini, 1995] cité page 1, 39
Gary Marchionini. *Information Seeking In Electronic Environments*. Cambridge University Press, 1995.

[Maron and Kuhns, 1960] cité page 25
M. E. Maron and J. L. Kuhns. On relevance, probabilistic indexing and information retrieval. *J. ACM*, 7(3) :216–244, 1960.

[Marteau *et al.*, 1999] cité page 186
P.-F. Marteau, C. de Loupy, P. Bellot, and M. El-Bèze. Systèmes & sécurité. *Le traitement automatique du langage naturel, outil d'assistance à la fonction d'intelligence économique : vers une architecture Push-Pull d'accès à l'information*, 5(4) :8–41, 1999.

[Maulik and Bandyopadhyay, 2000] cité page 98, 216, 216
U. Maulik and S. Bandyopadhyay. Genetic algorithm-based clustering technique. *Pattern Recognition*, 33 :1455–1465, 2000.

[Mckeown *et al.*, 1999] cité page 3, 229
Kathleen R. Mckeown, Judith L. Klavans, Vasileios Hatzivassiloglou, Regina Barzilay, and Eleazar Eskin. Towards multidocument summarization by reformulation : Progress and prospects. In *Proceedings of AAAI-99*, pages 453–460, 1999.

[McKeown *et al.*, 2002] cité page 49, 49
Kathleen R. McKeown, Regina Barzilay, David Evans, Vasileios Hatzivassiloglou, Judith L. Klavans, Ani Nenkova, Carl Sable, Barry Schiffman, and Sergey Sigelman. Tracking and summarizing news on a daily basis with Columbia's Newsblaster. In *Proc. Human Language Technology Conference*, 2002.

[Mendes and Sacks, 2003] cité page 176
M. E. S. Mendes and L. Sacks. Evaluating fuzzy clustering for relevance-based information access. In *The IEEE International Conference on Fuzzy Systems FUZZ-IEEE'03*, pages 648–653, 2003.

[Mihaila, 1996] cité page 37
George A. Mihaila. *WebSQL—A SQL-like Query Language for the World Wide Web.* master's thesis, Department of Computer Science, University of Toronto, 1996.

[Miller *et al.*, 1999] cité page 31
David R. H. Miller, Tim Leek, and Richard M. Schwartz. A hidden markov model information retrieval system. In *SIGIR '99 : Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 214–221, New York, NY, USA, 1999. ACM.

[Miller, 1995] cité page 39
G.A. Miller. Wordnet : A lexical database for english. *Communications of the ACM*, 38(11) :39–41, 1995.

[Mirkin, 1996] cité page 60, 63
B. Mirkin. *Mathematical Classification and Clustering.* Kluwer Academic Publishers, 1996.

[Moore and Pollack, 1992] cité page 105
Johanna D. Moore and Martha E. Pollack. A problem for rst : The need for multi-level discourse analysis. *Computational Linguistics*, 18 :537–544, 1992.

[Morris and Hirst, 1991] cité page 104, 106, 108, 128
J. Morris and G. Hirst. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1) :21–48, 1991.

[Murray, 1972] cité page 55
Daniel Mcclure Murray. *Document retrieval based on clustered files.* PhD thesis, Cornell University, Ithaca, NY, USA, 1972.

[Nakatani *et al.*, 1995] cité page 108
C.H. Nakatani, B.J. Grosz, D.D. Anh, and J. Hirschberg. Instructions for annotating discourses. Technical report, Technical Report TR-25-95, Harvard University Center for Research in Computing Technology, Cambridge, MA, USA, 1995.

[Nie and Brisebois, 1996] cité page 19
Jian-Yun Nie and Martin Brisebois. An inferential approach to information retrieval and its implementation using a manual thesaurus. *Artif. Intell. Rev.*, 10(5-6) :409–439, 1996.

[Nowell *et al.*, 1996] cité page 44
Lucy Terry Nowell, Robert K. France, Deborah Hix, Lenwood S. Heath, and Edward A. Fox. Visualizing search results : some alternatives to query-document similarity. In *SIGIR '96 : Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 67–75, New York, NY, USA, 1996. ACM.

[Over, 1998] cité page 69
Paul Over. Trec-6 interactive track report. In *Proceedings of the 6th Text Retrieval Conference TREC-6*, pages 57–64, 1998.

[Paisley and Parker, 1965] cité page 39
W. J. Paisley and E. B. Parker. Information retrieval as a receiver-controlled communication system. In *Education for Information Science*, pages 23–31. McMillan., London, UK, 1965.

[Papadimitriou, 1993] cité page 84
Christos H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1993.

[Passonneau, 1993] cité page 106
Rebecca J. Passonneau. Intention-based segmentation : Human reliability and correlation with linguistic cues. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 148–155, 1993.

[Pathak *et al.*, 2000] cité page 97
Praveen Pathak, Michael D. Gordon, and Weiguo Fan. Effective information retrieval using genetic algorithms based matching functions adaptation. In *HICSS*, 2000.

[Pédauque, 2003] cité page 104
Roger T. Pédauque. Document : forme, signe, médium, les reformulations du numérique. Technical report, STIC-CNRS, 2003.

[Perona and Malik, 1990] cité page 129
P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(7) :629–639, 1990.

[Pevzner and Hearst, 2002] cité page 111, 111, 112, 130, 134
L. Pevzner and M.A. Hearst. A critique and improvement of an evaluation metric for text segmentation. *Comput. Linguist.*, 28(1) :19–36, 2002.

[Picarougne, 2004] cité page 12, 18, 21, 44, 96, 98, 232
F. Picarougne. *Recherche d'information sur Internet par algorithmes évolutionnaires*. PhD thesis, Laboratoire d'Informatique, Université de Tours, novembre 2004.

[Pinon *et al.*, 1997] cité page 13
Jean-Marie Pinon, Sylvie Calabretto, and Line Poullet. Document semantic model : an experiment with patient medical records. In John Smith, editor, *ELPUB*, 1997.

[Pirolli *et al.*, 1996] cité page 43
Peter Pirolli, Patricia Schank, Marti Hearst, and Christine Diehl. Scatter/gather browsing communicates the topic structure of a very large text collection. In *CHI '96 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 213–220, New York, NY, USA, 1996. ACM.

[Platt, 1999] cité page 48
John C. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in kernel methods : support vector learning*, pages 185–208. MIT Press, Cambridge, MA, USA, 1999.

[Poli *et al.*, 2008] cité page 89
Riccardo Poli, William B. Langdon, and Nicholas Freitag McPhee. *A Field Guide to Genetic Programming*. Lulu Enterprises, UK Ltd, London, UK, 2008.

[Ponte and Croft, 1998] cité page 29, 30
Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *SIGIR '98 : Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, New York, NY, USA, 1998. ACM.

[Porter, 1980] cité page 14, 45
M.F. Porter. An algorithm for suffix stripping. *Program*, 14(3) :130–137, 1980.

[Preece, 1973] cité page 50
S. E. Preece. Clustering as an output option. In *Proceedings of the American Society for Information Science*, volume 10, pages 189–190, 1973.

[Qiu and Frei, 1993] cité page 39
Yonggang Qiu and Hans-Peter Frei. Concept-based query expansion. In *Proceedings of SIGIR-93, 16th ACM International Conference on Research and Development in Information Retrieval*, pages 160–169, Pittsburgh, US, 1993.

[Raghavan and Agarwal, 1987] cité page 97
Vijay Raghavan and Brijesh Agarwal. Optimal determination of user-oriented clusters : an application for the reproductive plan. In *Proceedings of the Second International Conference on Genetic Algorithms and their application*, pages 241–246, Mahwah, NJ, USA, 1987. Lawrence Erlbaum Associates, Inc.

[Raghavan and Wong, 1985] cité page 25
Vijay V. Raghavan and S. K. M. Wong. A critical analysis of vector space model for information retrieval. *Journal of the American Society for Information Science*, 37(5) :279–287, 1985.

[Rasmussen, 1992] cité page 59
Edie M. Rasmussen. Clustering algorithms. In *Information Retrieval : Data Structures & Algorithms*, pages 419–442. Prentice Hall, 1992.

[Reeves, 1993] cité page 84
Colin R. Reeves, editor. *Modern heuristic techniques for combinatorial problems*. John Wiley & Sons, Inc., New York, NY, USA, 1993.

[Reynar, 2000] cité page 128
J.C. Reynar. *Topic Segmentation : Algorithms and applications*. PhD thesis, University of Pennsylvania, Seattle, WA, 2000.

[Riloff, 1995] cité page 15
Ellen Riloff. Little words can make a big difference for text classification. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval*, pages 130–136, Seattle, US, 1995. ACM Press, New York, US.

[Robertson and Jones, 1988] cité page 28
Stephen E. Robertson and Karen Sparck Jones. Relevance weighting of search terms. In *Document retrieval systems*, pages 143–160. Taylor Graham Publishing, London, UK, UK, 1988.

[Robertson *et al.*, 1981] cité page 28
S. E. Robertson, C. J. Van Rijsbergen, and M. F. Porter. Probabilistic models of indexing and searching. In *SIGIR '80 : Proceedings of the 3rd annual ACM conference on Research and development in information retrieval*, pages 35–56, Kent, UK, UK, 1981. Butterworth & Co.

[Robertson *et al.*, 1992] cité page 28, 31, 32, 33, 33, 153
Stephen E. Robertson, Steve Walker, Micheline Hancock-Beaulieu, Aarron Gull, and Marianna Lau. Okapi at TREC. In *Text REtrieval Conference*, pages 21–30, 1992.

[Robertson *et al.*, 1993] cité page 33, 153
Stephen E. Robertson, Steve Walker, S. Jones, Micheline M. Hancock-beaulieu, and M. Gatford. Okapi at trec-2. In *In The Second Text REtrieval Conference (TREC-2), NIST Special Special Publication 500-215*, pages 21–34, 1993.

[Robertson *et al.*, 1995] cité page 33, 153
Stephen E. Robertson, Steve Walker, S. Jones, Micheline M. Hancock-beaulieu, and M. Gatford. Okapi at trec-3. In *The Third Text REtrieval Conference (TREC-3)*, pages 109–126, 1995.

[Rocchio, 1966] cité page 54
J.J. Rocchio. *Document retrieval systems - Optimization and evaluation*. PhD thesis, Harvard Computation Laboratory, 1966.

[Rocchio, 1971] cité page 40
J. J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System : Experiments in Automatic Document Processing* [1971b], pages 313–323.

[Rorvig, 1999] cité page 53
Mark Rorvig. Images of similarity : a visual exploration of optimal similarity metrics and scaling properties of trec topic-document sets. *Journal of the American Society of Information Science*, 50(8) :639–651, 1999.

[Ruthven and Lalmas, 2003] cité page 39
Ian Ruthven and Mounia Lalmas. A survey on the use of relevance feedback for information access systems. *Knowl. Eng. Rev.*, 18(2) :95–145, 2003.

[Ruthven *et al.*, 2001] cité page 12, 39
Ian Ruthven, Anastasios Tombros, and Joemon M. Jose. A study on the use of summaries and summary-based query expansion for a question-answering task. In U. Thiel and N. Fuhr, editors, *23rd BCS European Annual Colloquium on Information Retrieval Research (ECIR 2001)*, pages 1–14. Electronic Workshops in Computing, 2001.

[Salton and Buckley, 1988] cité page 17, 17, 52
Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5) :513–523, 1988.

[Salton and Buckley, 1994] cité page 186
Gerard Salton and Christopher Buckley. Automatic structuring and retrieval of large text files. *Communications of the ACM*, 37(2) :97–108, 1994.

[Salton and Buckley, 1997] cité page 12
Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. In *Readings in information retrieval*, pages 355–364. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[Salton and McGill, 1986] cité page 16, 20, 24
Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[Salton *et al.*, 1974] cité page 16
Gerard Salton, C. S. Yang, and Clement T. Yu. A theory of term importance in automatic text analysis. Technical report, Cornell University, Ithaca, NY, USA, 1974.

[Salton *et al.*, 1983] cité page 21, 21, 22, 22
Gerard Salton, Edward A. Fox, and Harry Wu. Extended boolean information retrieval. *Commun. ACM*, 26(11) :1022–1036, 1983.

[Salton *et al.*, 1993] cité page 151
G. Salton, J. Allan, and C. Buckley. Approaches to passage retrieval in full text information systems. In *Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 49–58, Pittsburgh, Pennsylvania, USA, 1993. ACM.

[Salton *et al.*, 1996] cité page 4, 104, 107, 114, 142
Gerard Salton, Amit Singhal, Chris Buckley, and Mandar Mitra. Automatic text decomposition using text segments and text themes. In *The Seventh ACM Conference on Hypertext (Hypertext'96)*, pages 53–65, Washington, DC, USA, 1996. ACM.

[Salton, 1962] cité page 18
Gerard Salton. Manipulation of trees in information retrieval. *Communication of the ACM*, 5(2) :103–114, 1962.

[Salton, 1968] cité page 1
Gerard. Salton. *Automatic Information Organization and Retrieval.* McGraw Hill Text, 1968.

[Salton, 1971a] cité page 49, 51, 55
Gerard Salton. Cluster search strategies and the optimization of retrieval effectiveness. In *The SMART Retrieval System : Experiments in Automatic Document Processing* [1971b], pages 223–242.

[Salton, 1971b] cité page 23, 31, 55, 266, 268
Gerard Salton. *The SMART Retrieval System : Experiments in Automatic Document Processing.* Prentice-Hall, 1971.

[Sanderson, 1997] cité page 15
Mark Sanderson. *Word Sense Disambiguation and Information Retrieval.* PhD thesis, University of Glasgow, UK, 1997.

[Savoy, 1993] cité page 15
Jacques Savoy. Stemming of french words based on grammatical categories. *Journal of the American Society for Information Science*, 44(1) :1–9, 1993.

[Schauble, 1998] cité page 69
Peter Schauble. Cross-language information retrieval (clir) track overview. In *Proceedings of the 6th Text Retrieval Conference TREC-6*, 1998.

[Schlieder and Meuss, 2002] cité page 104
Torsten Schlieder and Holger Meuss. Querying and ranking xml documents. *J. Am. Soc. Inf. Sci. Technol.*, 53(6) :489–503, 2002.

[Schott, 1995] cité page 92
Jason R. Schott. Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, May 1995.

[Schütze and Silverstein, 1997] cité page 59
Hinrich Schütze and Craig Silverstein. Projections for efficient document clustering. In *SIGIR '97 : Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 74–81, New York, NY, USA, 1997. ACM.

[Schwefel, 1981] cité page 87
Hans-Paul Schwefel. *Numerical Optimization of Computer Models.* John Wiley & Sons, Inc., New York, NY, USA, 1981.

[Sebrechts *et al.*, 1999] cité page 44
Marc M. Sebrechts, John Cugini, Sharon J. Laskowski, Joanna Vasilakis, and

Michael S. Miller. Visualization of search results : A comparative evaluation of text, 2d, and 3d interfaces. In *Research and Development in Information Retrieval*, pages 3–10, 1999.

[Shannon, 1948] cité page 16
Claude Shannon. A mathematical theory of communication. *Bell system technical journal*, 27 :379–423 and 623–656, 1948.

[Shaw, 1993] cité page 52
W. M. Shaw. Controlled and uncontrolled subject descriptions in the cf database : a comparison of optimal cluster-based retrieval results. *Inf. Process. Manage.*, 29(6) :751–763, 1993.

[Shneiderman, 1992] cité page 41
Ben Shneiderman. Tree visualization with tree-maps : 2-d space-filling approach. *ACM Trans. Graph.*, 11(1) :92–99, 1992.

[Silla *et al.*, 2004] cité page 98
Carlos Nascimento Silla, Gisele L. Pappa, Alex Alves Freitas, and Celso A. A. Kaestner. Automatic text summarization with genetic algorithm-based attribute selection. In Christian Lemaître, Carlos A. Reyes, and Jesús A. González, editors, *IBERAMIA*, volume 3315 of *Lecture Notes in Computer Science*, pages 305–314. Springer, 2004.

[Silverstein and Pedersen, 1997] cité page 78, 189
Craig Silverstein and Jan O. Pedersen. Almost-constant-time clustering of arbitrary corpus subsets. In *SIGIR*, pages 60–66. ACM, 1997.

[Silverstein *et al.*, 1998] cité page 41
Craig Silverstein, Monika Henzinger, Hannes Marais, and Michael Moricz. Analysis of a very large altavista query log. Technical Report 1998-014, Digital SRC, 1998.

[Singhal *et al.*, 1996a] cité page 32, 33, 33, 33, 33, 35, 35
Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *Research and Development in Information Retrieval*, pages 21–29, 1996.

[Singhal *et al.*, 1996b] cité page 4, 31, 31, 32, 32, 33, 33, 34, 151, 153, 153, 157, 157
Amit Singhal, Gerard Salton, Mandar Mitra, and Chris Buckley. Document length normalization. *Inf. Process. Manage.*, 32(5) :619–633, 1996.

[Sitbon and Bellot, 2005] cité page 106, 116
L. Sitbon and P. Bellot. Segmentation thématique par chaînes lexicales pondérées. In *TALN*, 2005.

[Skorochod'ko, 1971] cité page 106
E. F. Skorochod'ko. Adaptive method of automatic abstracting and indexing. In *IFIP Congress (2)*, pages 1179–1182, 1971.

[Sneath and Sokal, 1973] cité page 52, 52
P.H.A. Sneath and R.R. Sokal. *Numerical taxonomy : the principles and practice of numerical classification.* W.H. Freeman, San Francisco, USA, 1973.

[Song and Croft, 1999] cité page 31
Fei Song and W. Bruce Croft. A general language model for information retrieval. In *CIKM '99 : Proceedings of the eighth international conference on Information and knowledge management*, pages 316–321, New York, NY, USA, 1999. ACM.

[Sparck-Jones and van Rijsbergen, 1975] cité page 71
K. Sparck-Jones and C. van Rijsbergen. Report on the need for and provision of an « ideal » information retrieval test collection. Technical report, British Library Research and Development Report 5266, Computer Laboratory, University of Cambridge, Cambridge, UK, 1975.

[Spink *et al.*, 2001] cité page 41
Amanda Spink, Dietmar Wolfram, Major B. J. Jansen, and Tefko Saracevic. Searching the web : the public and their queries. *J. Am. Soc. Inf. Sci. Technol.*, 52(3) :226–234, 2001.

[Stanfill and Waltz, 1992] cité page 151
Craig Stanfill and David L. Waltz. Statistical methods, artificial intelligence, and information retrieval. In *Text-based intelligent systems : current research and practice in information extraction and retrieval*, pages 215–225. Lawrence Erlbaum Associates, Inc., Mahwah, NJ, USA, 1992.

[Stein *et al.*, 2000] cité page 230
Gees C. Stein, Amit Bagga, and G. Bowden Wise. Multi-document summarization : Methodologies and evaluations. In *Proceedings of the 7th Conference on Automatic Natural Language Processing (TALN'00)*, pages 337–346, 2000.

[Stewart, 1993] cité page 28
G. W. Stewart. On the early history of the singular value decomposition. *SIAM Rev.*, 35(4) :551–566, 1993.

[Stokoe *et al.*, 2003] cité page 15
Christopher Stokoe, Michael P. Oakes, and John Tait. Word sense disambiguation in information retrieval revisited. In *SIGIR '03 : Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 159–166, New York, NY, USA, 2003. ACM.

[Swan and Allan, 1998] cité page 44
Russell C. Swan and James Allan. Aspect windows, 3-d visualizations, and indirect comparisons of information retrieval systems. In *Research and Development in Information Retrieval*, pages 173–181, 1998.

[Taylor, 1968] cité page 39
Robert S. Taylor. Question negotiation and information seeking in libraries. *Journal of College and Research Libraries*, 29(3) :178–194, 1968.

[Tibshirani *et al.*, 2000] cité page 59
R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a dataset via the gap statistic. Technical report, Technical Report 208, Departement of Statistics, Stanford University, 2000.

[Tombros and Sanderson, 1998] cité page 64
Anastasios Tombros and Mark Sanderson. Advantages of query biased summaries in information retrieval. In *SIGIR '98 : Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 2–10, New York, NY, USA, 1998. ACM.

[Tombros and Van Rijsbergen, 2004] cité page 6, 188, 188
Anastasios Tombros and C. J. Van Rijsbergen. Query-sensitive similarity measures for information retrieval. *Knowledge Information Systems*, 6(5) :617–642, 2004.

[Tombros *et al.*, 2002] cité page 1, 77, 80, 81, 177, 177, 178, 178, 229
Anastasios Tombros, Robert Villa, and C. J. Van Rijsbergen. The effectiveness of query-specific hierarchic clustering in information retrieval. *Information Processing & Management*, 38(4) :559–582, 2002.

[Tombros *et al.*, 2003] cité page 6, 188
Anastasios Tombros, Joemon M. Jose, and Ian Ruthven. Clustering top-ranking sentences for information access. In Traugott Koch and Ingeborg Sølvberg, editors, *ECDL*, volume 2769 of *Lecture Notes in Computer Science*, pages 523–528. Springer, 2003.

[Tombros, 2002] cité page 12, 50, 52, 55, 56, 63, 64, 65, 173, 188
Anastasios Tombros. *The Effectiveness of Query-Based Hierarchic Clustering of Documents for Information Retrieval*. PhD thesis, University of Glasgow, UK, 2002.

[Tou and Gonzalez, 1974] cité page 59
J. T. Tou and R. C. Gonzalez. *Pattern recognition principles*. Applied Mathematics and Computation, Reading, Mass. : Addison-Wesley, 1974, 1974.

[TREC, 2006] cité page 33, 33
TREC. Text retrieval conference, janvier 2006. http ://trec.nist.gov.

[Tufféry, 2005] cité page 54
Stephane Tufféry. *Data mining et statistique décisionnelle : l'intelligence dans les bases de données*. Technip, 2005.

[Turtle and Croft, 1989] cité page 26
Howard Turtle and W. Bruce Croft. Inference networks for document retrieval. In *Proc. SIGIR*, pages 1–24. ACM Press, 1989.

[Tversky, 1977] cité page 173
Amos Tversky. Features of similarity. *Psychological Review*, 84(2) :327–352, 1977.

[Utiyama and Isahara, 2001] cité page 106, 106
M. Utiyama and H. Isahara. A statistical model for domain-independent text

segmentation. In *Meeting of the Association for Computational Linguistics*, pages 491–498, 2001.

[Vaishnavi, 1989] cité page 18
Vijay K. Vaishnavi. Multidimensional balanced binary trees. *IEEE Trans. Comput.*, 38(7) :968–985, 1989.

[Van Rijsbergen *et al.*, 1981] cité page 51
C.J. Van Rijsbergen, D. J. Harper, and M. F. Porter. The selection of good search terms. *Information Processing & Management*, 7 :77–91, 1981.

[Van Rijsbergen, 1979] cité page 13, 16, 16, 18, 25, 26, 49, 53, 55, 56, 60, 73, 73, 76, 186
C.J. Van Rijsbergen. *Information Retrieval, 2nd edition*. Departement of Computer Science, Information Retrieval Group, University of Glasgow, 1979.

[Veerasamy and Belkin, 1996] cité page 44
Aravindan Veerasamy and Nicholas J. Belkin. Evaluation of a tool for visualization of information retrieval results. In *SIGIR '96 : Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 85–92, New York, NY, USA, 1996. ACM.

[Veerasamy and Heikes, 1997] cité page 44
Aravindan Veerasamy and Russell Heikes. Effectiveness of a graphical display of retrieval results. *SIGIR Forum*, 31(SI) :236–245, 1997.

[Voorhees and Harman, 1997] cité page 1, 25, 41
Ellen M. Voorhees and Donna Harman. Overview of the fifth text retrieval conference (trec-5). In *Proceedings of the Fifth Text Retrieval Conference*, pages 1–28. NIST Special Publication 500-238, 1997.

[Voorhees, 1986] cité page 77, 173
Ellen Marie Voorhees. *The effectiveness and efficiency of agglomerative hierarchic clustering in document retrieval*. PhD thesis, Cornell University, Ithaca, NY, USA, 1986.

[Voorhees, 1993] cité page 15
Ellen M. Voorhees. Using wordnet to disambiguate word senses for text retrieval. In *SIGIR '93 : Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 171–180, New York, NY, USA, 1993. ACM.

[Voorhees, 1994] cité page 39
Ellen M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR '94 : Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 61–69, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[Voorhees, 2002] cité page 68
Ellen M. Voorhees. The philosophy of information retrieval evaluation. In *Proceedings of the The Second Workshop of the Cross-Language Evaluation Forum*

on Evaluation of Cross-Language Information Retrieval Systems, pages 355–370. Springer-Verlag, 2002.

[Voorneveld, 2003] cité page 90
Mark Voorneveld. Characterization of pareto dominance. *Operations Research Letters*, 31(1) :7–11, 2003.

[Vrajitoru, 1998] cité page 97
Dana Vrajitoru. Crossover improvement for the genetic algorithm in information retrieval. *Inf. Process. Manage.*, 34(4) :405–415, 1998.

[Walker, 1992] cité page 106
Marilyn A. Walker. Redundancy in collaborative dialogue. In *Proceedings of the 14th conference on Computational linguistics*, pages 345–351, Morristown, NJ, USA, 1992. Association for Computational Linguistics.

[Washburne, 1927] cité page 41
J. N. Washburne. An experimental study of various graphic, tabular and textual methods of presenting quantitative material. *Journal of Educational Psychology*, 18(6) :361–367, 1927.

[Wei *et al.*, 2008] cité page 12
Furu Wei, Wenjie Li, Qin Lu, and Yanxiang He. A cluster-sensitive graph model for query-oriented multi-document summarization. In Craig Macdonald, Iadh Ounis, Vassilis Plachouras, Ian Ruthven, and Ryen W. White, editors, *ECIR*, volume 4956 of *Lecture Notes in Computer Science*, pages 446–453. Springer, 2008.

[Willett, 1983] cité page 53, 53
Peter Willett. Similarity coefficients and weighting functions for automatic document classification : an empirical comparison. *International Classification*, 3 :138–142, 1983.

[Willett, 1988] cité page 48, 53, 56, 60, 63
Peter Willett. Recent trends in hierarchic document clustering : a critical review. *Information Processing & Management*, 24(5) :577–597, 1988.

[Winograd, 1972] cité page 12
T. Winograd. *Understanding Natural Language*. Edinburgh University Press, 1972.

[Wise, 1999] cité page 52
James A. Wise. The ecological approach to text visualization. *Journal of the American Society of Information Science*, 50(13) :1224–1233, 1999.

[Wong *et al.*, 1985] cité page 25
S. K. M. Wong, Wojciech Ziarko, and Patrick C. N. Wong. Generalized vector spaces model in information retrieval. In *SIGIR '85 : Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 18–25, New York, NY, USA, 1985. ACM.

[Yaari, 1997] cité page 105
Yaakov Yaari. Segmentation of expository texts by hierarchical agglomerative clustering. *CoRR*, cmp-lg/9709015, 1997.

[Yang, 1993] cité page 97
J.J. Yang. *Use of Genetic Algorithms for Query Improvement in Information Retrieval Based on a Vector Space Model*. Phd. Thesis, University of Pittsburgh, Pittsburgh, PA, 1993.

[Yarowsky, 1995] cité page 15
David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Meeting of the Association for Computational Linguistics*, pages 189–196, 1995.

[Yule, 1897] cité page 159, 160
G. Udny Yule. On the theory of correlation. *Journal of the Royal Statistical Society*, 60(4) :812–854, December 1897.

[Zadeh, 1965] cité page 21
Lofti A. Zadeh. Fuzzy sets. *Information Control*, 8 :338–353, 1965.

[Zamir and Etzioni, 1998] cité page 43, 58, 115
Oren Zamir and Oren Etzioni. Web document clustering : a feasibility demonstration. In *SIGIR '98 : Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 46–54, New York, NY, USA, 1998. ACM.

[Zamir and Etzioni, 1999] cité page 43, 49, 50
Oren Zamir and Oren Etzioni. Grouper : a dynamic clustering interface to web search results. In *WWW '99 : Proceedings of the eighth international conference on World Wide Web*, pages 1361–1374, New York, NY, USA, 1999. Elsevier North-Holland, Inc.

[Zhao *et al.*, 2004] cité page 29
Bing Zhao, Matthias Eck, and Stephan Vogel. Language model adaptation for statistical machine translation with structured query models. In *COLING '04 : Proceedings of the 20th international conference on Computational Linguistics*, page 411, Morristown, NJ, USA, 2004. Association for Computational Linguistics.

[Zipf, 1949] cité page 15
George K. Zipf. *Human Behavior and the Principle of Least Effort*. Addison-Wesley (Reading MA), 1949.

[Zitzler, 1999] cité page 6, 92, 92, 93, 122
E. Zitzler. *Evolutionary Algorithms for Multiobjective Optimization : Methods and Applications*. Phd thesis, Swiss Federal Institute of Technology (ETH) Zurich, December 1999.

[Zobel and Moffat, 1998] cité page 17, 17
Justin Zobel and Alistair Moffat. Exploring the similarity space. *SIGIR Forum*, 32(1) :18–34, 1998.

[Zobel *et al.*, 1995] cité page 151
Justin Zobel, Alistair Moffat, Ross Wilkinson, and Ron Sacks-Davis. Efficient retrieval of partial documents. *Information Processing and Management*, 31(3) :361–377, 1995.

[Zobel, 1998] cité page 71, 155, 155
Justin Zobel. How reliable are the results of large-scale information retrieval experiments ? In *SIGIR '98 : Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 307–314, New York, NY, USA, 1998. ACM.

Vers la conception de documents composites : extraction et organisation de l'information pertinente

Résumé

Au cours de ces dernières années, le domaine de la recherche d'information s'est élargi à la mise en place d'applications ne visant plus uniquement à aider l'utilisateur dans sa tâche de localisation des documents pertinents, mais cherchant à lui construire une réponse synthétique permettant de satisfaire ses besoins en information. Dans ce contexte, cette thèse se concentre sur la production d'une entité, appelée document composite, représentant un aperçu des différents types d'information que l'utilisateur pourra trouver, en rapport avec sa requête, dans le corpus interrogé. Après s'être interrogés sur le mode d'extraction et de sélection des fragments de texte à faire figurer dans ce document composite, l'étude réalisée nous a finalement conduits à la mise en place d'un algorithme multi-objectifs, de recherche du sous-ensemble de segments thématiques maximisant conjointement un critère de proximité à la requête et un critère de représentativité des thématiques abordées par les documents considérés. Outre la conception du document composite qui est l'objectif central de cette thèse, les contributions réalisées concernent le découpage des documents et son évaluation, les mesures de pertinence et de similarité des textes, l'impact que peut avoir l'individualisation des thématiques en recherche d'information, le mode d'évaluation des systèmes utilisant un clustering des résultats et enfin, la prise en considération de la requête dans les processus de clustering.

Mots-clés : Recherche d'information, document composite, segmentation thématique, Passage Retrieval, catégorisation, optimisation multi-objectif

Toward composite documents conception : extraction and organisation of relevant informations

Abstract

In recent years, information retrieval has expanded its area to the development of applications whose purpose is not solely to help the user to locate the relevant documents, but also try to build a synthetic answer as response to his expressed information needs. In this context, this thesis focuses on the production of an entity, called composite document, representing an overview of the different types of information that the user can find, in connection with his request, in the corpus in concern. After being concerned about the method of extraction and selection of fragments of text to be included in the composite document, the study has finally led to the setting up of a multi-objective algorithm, which aims at finding the thematic segments subset maximizing two criteria of query proximity and thematic representativeness. Beyond the composite document conception, the realized contributions concern the thematic segmentation and its evaluation, the relevance estimations and similarity computations, the impact of the thematic individualization in the field of information retrieval, the evaluation of systems presenting search results in term of a clusters set and, at last, the ways of query consideration in texts clustering process.

Keywords : Information Retrieval, composite document, thematic segmentation, Passage Retrieval, clustering, multi-objective optimisation