L'équipe du GRDS au Défi Fouille de textes 2006 : Indexo-II

Lyne Da Sylva*, Graham Russell**, Yves Marcoux*, Frédéric Doll***

*GRDS, École de bibliothéconomie et des sciences de l'information
{Lyne.Da.Sylva, Yves.Marcoux}@UMontreal.CA
http://mapageweb.umontreal.ca/{dasylval, marcoux}
** GRDS, École de bibliothéconomie et des sciences de l'information
graham.j.russell @sympatico.ca
***Convera
fdoll@convera.com

Résumé. Cet article présente les travaux de l'équipe du GRDS à la compétition DEFT'06. On y décrit un algorithme de segmentation automatique de texte qui utilise plusieurs informations relatives à la distribution des mots dans le texte pour proposer des frontières candidates : cohésion lexicale, chaînes lexicales, taux d'introduction de vocabulaire. Un arbre de décision est déterminé pour chaque corpus d'entraînement. Les résultats de précision obtenus sont en moyenne de 58,04% (largement au-dessus de la moyenne des équipes participantes) alors que le rappel est à 25,59%. Pour un des corpus, les résultats sont supérieurs à la moyenne (f-score de 59,20%, comparé à 21,14% en moyenne). L'algorithme est contrasté avec un autre, utilisé dans un système d'indexation automatique de monographies, afin d'illustrer l'impact de la finalité de la segmentation sur la tâche elle-même.

1 Introduction

Bien que la segmentation automatique de texte puisse être considérée comme une fin en soi, elle n'est le plus souvent qu'une partie d'un traitement plus complexe. Par exemple, la segmentation peut intervenir comme première étape dans l'indexation automatique (attribution de termes descripteurs aux différentes parties d'un texte). C'est d'ailleurs dans le contexte de l'indexation automatique de monographies qu'est né l'intérêt de l'équipe GRDS pour la segmentation automatique. Certains aspects de nos travaux sur l'indexation automatique (Da Sylva 2006, Marcoux et al. 2005) ont pu être récupérés pour la présente implémentation, mais nous avons constaté que la finalité de la segmentation (la raison pour laquelle elle est effectuée) conditionne fortement le choix de l'approche à utiliser. C'est pourquoi la présente implémentation diffère de celles utilisées dans nos travaux antérieurs sur l'indexation automatique. En plus de présenter l'algorithme développé, nous tentons dans cet article de mettre en lumière les liens entre la finalité de la segmentation et l'approche à utiliser.

2 Travaux reliés

Yaari (1997) reconnaît deux types de segmentation de texte : l'approche basée sur la cohésion lexicale et celle qui utilise des informations de plusieurs sources. Notre implémentation relève de cette dernière : elle est inspirée des travaux de TextTiling de Hearst (1997).

2.1 Segmentation par cohésion lexicale

Comme Hearst (1997) et d'autres, nous nous basons en large partie sur des indicateurs de cohésion présents dans le texte, notamment la cohésion lexicale et grammaticale. La cohésion fait référence à la propriété qu'a un texte donné d'exprimer de manière explicite des « relations sémantiques entre les phrases d'un discours réalisées par des éléments de niveau lexico-grammatical; soit des mots » (Patry et al., 1989 ; voir aussi Halliday et Hasan, 1976). La cohésion lexicale est exprimée par des mots provenant des classes ouvertes (noms, verbes, adjectifs) tandis que la cohésion grammaticale fait appel à des mots des classes fermées (pronoms, déterminants, etc.). Notre algorithme utilise le premier type.

Les marqueurs de cohésion comprennent notamment les suivants (dont seulement certains, cependant, sont pris en compte par les algorithmes de segmentation automatique) :

- répétition exacte de chaînes de caractères (par exemple, la répétition du mot "cohésion" dans les paragraphes ci-dessus)
- co-occurrence de variantes du même lemme (cheval / chevaux)
- utilisation de mots avec radical commun (construction / construire; nation / national / nationaliser / nationalisation)
- utilisation de synonymes, hyperonymes (humain / personne ; caniche / chien)
- pronoms personnels (Elle, …)
- déterminants démonstratifs ou possessifs (Pour ces expériences, … Leurs idées …)
- certains adverbes ou conjonctions (en effet, par contre, ainsi, …)

La segmentation basée sur la cohésion repose sur l'hypothèse que des passages cohérents d'un point de vue thématique partagent un vocabulaire commun. Elle opère simplement : en comptant le nombre de mots en commun ou d'autres moyens cohésifs dans deux phrases ou segments successifs, on peut calculer un *score de cohésion* pour chaque unité textuelle. Ces scores peuvent être reportés sur un graphique, indiquant des sommets et des vallées ; ces vallées correspondent aux coupures possibles pour la segmentation (ou *frontières candidates* – voir la figure 1). En découpant le texte au point des vallées significatives, on obtient une suite de passages supposément cohésifs, donc cohérents d'un point de vue thématique. C'est essentiellement l'approche retenue dans Hearst (1997). Des travaux subséquents ont porté sur la question du lissage de la courbe ou la détermination du seuil minimum qui déclenche une coupure. Morris et Hirst (1991) utilisent un thésaurus dans la mesure de similarité, afin que les synonymes et hyperonymes contribuent à augmenter le score de similarité.

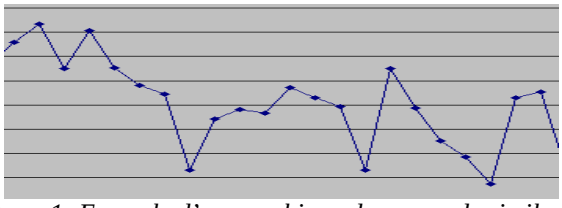

FIGURE 1: Exemple d'un graphique de scores de similarité

2.2 Apprentissage-machine

Pour la tâche de DEFT'06, les trois types de textes étant qualitativement différents, une approche homogène s'est avérée peu satisfaisante. Par contre, nous disposions de corpus

volumineux : il était donc envisageable de profiter de l'abondance de données afin d'entraîner un algorithme d'apprentissage-machine à repérer les changements de segments en fonction des caractéristiques de chaque texte. Ainsi, un système basé sur les arbres de décision (Murthy et al., 1994) a été incorporé à notre algorithme.

3 Approche utilisée

3.1 Introduction

La segmentation est effectuée en utilisant une combinaison de critères reflétant les propriétés du texte qui sont pertinentes pour la structure thématique ; la plupart relèvent de patrons de distribution lexicale. Chaque frontière potentielle de segment est associée à un vecteur de scores numériques représentant les valeurs locales de ces propriétés. Pour les données d'entraînement, chaque vecteur a été annoté à l'aide de sa catégorie (frontière ou non-frontière) et transmis à un arbre de décision. Pour les données test, les vecteurs non annotés ont été évalués par l'arbre de décision ainsi créé, pour produire les catégories résultantes.

3.2 Prétraitement

Le prétraitement des textes est relativement simple : chaque « phrase » du corpus de DEFT'06 est tokenisée, en séparant la chaîne de caractères sur l'espace et les caractères de ponctuation non ambigus ; les majuscules initiales sont remplacées par des minuscules et le résultat est soumis à une radicalisation (à l'aide d'une version modifiée du module Perl Lingua::Stem::Fr), puis filtré, afin de supprimer tout élément qui n'est pas un mot significatif (*content words*). La segmentation en phrases du corpus initial a été conservée.

Pour le corpus « Lois », le fait que les frontières de segments coïncident avec celles des articles est exploité en concaténant les phrases ; ainsi, les unités considérées par le système débutent toutes par la chaîne « Article X ». Ceci nécessite un post-traitement ad hoc pour réassigner correctement les identificateurs des phrases. Puisque certaines portions du corpus « Discours » ne sont pas accentuées, les accents sont retirés dans la totalité de ce texte.

3.3 Termes et pondération des termes

Tous les éléments qui subsistent après la phase de filtrage décrite ci-dessus sont traités comme des termes. Il n'y a ainsi aucun étiquetage en parties du discours ni reconnaissance d'expressions à plusieurs mots. Notons que le sens du mot terme ici est tel qu'il est utilisé dans les travaux en indexation et non en terminologie.

Les termes sont pondérés à l'aide du *idf* résiduel, ou RIDF (Church, 1995 ; Manning et Schütze, 1999, p. 553). Ceci constitue une mesure de l'irrégularité de la distribution statistique, qui consiste à comparer la valeur *idf* (*inverse document frequency*) d'un terme avec celle qui est prédite par le modèle de Poisson. Cette mesure a été proposée pour distinguer les mots significatifs des mots non significatifs (*non-content words*). Ici, la mesure est interprétée comme un indicateur de « thématicité », l'hypothèse sous-jacente étant qu'il devrait y avoir un degré de corrélation élevé entre un RIDF élevé et les éléments lexicaux les plus associés au thème courant du passage.

Comme la valeur *idf*, le RIDF est normalement calculé pour un terme donné sur la base de tous les documents d'une grande collection. Pour cette application, chacun des trois textes a été divisé en un nombre de pseudo-documents, dont la taille a été déterminée empiriquement. Concrètement, les valeurs RIDF de tous les termes ont été calculées pour des tailles variables de pseudo-documents (50, 100, 150, 200 et 250 termes) ; la taille donnant la variance la plus élevée pour la valeur RIDF par terme a été retenue.

3.4 Cohésion lexicale

Bien que le terme « cohésion lexicale » puisse, en principe, être appliqué à d'autres classes de critères de segmentation, il fait référence ici à une mesure locale de similarité textuelle basée sur le contenu lexical du contexte immédiat d'une frontière potentielle de segment. L'implémentation suit celle de Hearst (1997), utilisant un modèle vectoriel de sacs de mots, mais comporte quelques différences mineures notées ci-dessous.

Les vecteurs de termes pondérés sont construits en représentant les contextes gauche et droit de chaque frontière possible de segment. Plusieurs paramètres utilisés dans ce processus ont été déterminés empiriquement en utilisant la méthode décrite ci-dessous.

- les contextes
  - la taille en nombre de tokens
  - le fait que les contextes devraient être étendus à la frontière de segment la plus proche (précédente ou suivante)
- la valeur assignée à chaque terme
  - RIDF simple
  - le produit de RIDF et la fréquence dans le contexte
  - le produit de RIDF et la fréquence dans le corpus

Pour les trois corpus test, la meilleure performance a été obtenue en utilisant un contexte de base de 40 tokens, étendu au besoin pour contenir la phrase entière, et avec des éléments vectoriels pondérés à l'aide de la valeur RIDF du terme correspondant. De plus, la mesure du cosinus s'est révélée plus apte à donner de meilleurs résultats globaux que la mesure de la distance euclidienne. Les scores de cohésion sont convertis en profondeurs et lissés tel que décrit dans Hearst (op.cit.). Une fonction de lissage plus complète a été employée, cependant, puisqu'un lissage naïf basé sur la moyenne peut déplacer les minimas locaux de leur position réelle. Essentiellement, ceci implique d'abord de lisser de façon classique, en prenant la moyenne des valeurs voisines, et puis de déplacer toute valeur minimum *vm* trouvée à la position *i* à la position la moins élevée dans l'intervalle $i-k \ldots i+k$ dans la séquence originale, où *k* est fixé à 3 (cette valeur est évidemment liée à la taille du contexte dans lequel le calcul de moyenne est effectué).

3.5 Chaînes lexicales

Le deuxième critère employé dans le système Indexo-II met en jeu les chaînes lexicales (voir une présentation initiale des idées derrière cette notion dans Morris et Hirst, 1991 ainsi que dans Barzilay et Elhadad, 1997). Dans cette implémentation, en restant fidèle à l'approche assez simple de distribution lexicale présente à la section précédente, les chaînes sont

construites sur la base de la distribution des termes dans les phrases : plus simplement, il s'agit de l'ensemble de phrases dans lesquelles un terme apparaît.

Plus spécifiquement, une chaîne lexicale pour un terme *t* est la suite maximale de phrases contiguës $P_{t,i,k} = < p_i, p_{i+1}, \ldots p_k >$ telle que *t* apparaît au moins une fois dans p_1 et dans p_k, et peut-être dans certaines des p_j où i < j < k. Ainsi, c'est une suite de phrases dont la première et la dernière contiennent *t*, même si *t* est absent de certaines des phrases p_j entre les deux. La longueur (en nombre de phrases) de tout intervalle entre occurrences successives est limitée à ne pas dépasser un maximum δ. Il n'y a aucune chaîne à une seule phrase. On dit qu'une chaîne $P_{t,i,k}$ contient une phrase p_j si et seulement si i < j ≤ k. Par exemple, la distribution < 4, 6, 7, 10, 16, 22, 24, 25 > pour le terme *t* génère deux chaînes lexicales lorsque δ est fixé à 5, tel qu'indiqué au tableau 1 (avec l'ensemble des phrases contenus dans chacune).

$P_{t,4,10}$	$< p_4, p_5, p_6, p_7, p_8, p_9 >$
$P_{t,22,25}$	$< p_{22}, p_{23}, p_{24} >$

TABLEAU 1 - Deux chaînes lexicales et leur contenu

Avec un delta de 6 ou plus, une seule chaîne $P_{t,4,25}$ serait générée.

Quelques expérimentations ont été effectuées afin d'étudier les performances de différentes pondérations d'une chaîne $P_{t,i,k}$:

- uniforme (sans poids)
- RIDF de *t*
- « densité » de la chaîne, c'est-à-dire le ratio du nombre de liens avec la distance entre les extrémités de la chaîne : $|P_{t,i,k}|/(1+k-j)$
- le produit de RIDF et de la densité

Dans cette expérimentation, la valeur RIDF simple semblait donner les meilleurs résultats, mais la différence était très faible.

Cette pondération a été appliquée comme fonction de poids W_{CL} sur des paires constituées des indices de phrases et des chaînes lexicales, définie comme suit :

$$W_{CL}(j, P_{t,i,k}) \equiv \begin{cases} RIDF(t) & si\ i < j \le k \\ 0 & autrement \end{cases}$$

Ainsi, la pondération associée à une chaîne est assignée seulement aux phrases qui la contiennent. Le score lié aux chaînes lexicales pour les frontières candidates situées immédiatement avant la phrase p_j, $C(P_j)$, est simplement la somme des scores individuels des chaînes qui contiennent p_j. Cette valeur est calculée pour chaque frontière de segment candidate dans le texte.

$$C(p_j) = \sum_{t,i,k;\ i<j\le k} W_{CL}(j, P_{t,i,k})$$

3.6 Taux d'introduction du vocabulaire

Le troisième critère distributionnel utilisé ici tient compte des variations observées dans le taux d'introduction de termes nouveaux dans le texte. L'intuition qui motive ce critère est

qu'une augmentation de ce taux tend à s'accompagner de changements dans la thématique du texte, alors que des taux faibles tendent à s'observer vers la fin d'un segment. Bien sûr, cette hypothèse n'est pas valide de manière uniforme pour tous les types de textes. Certains styles d'écriture préconisent les allusions précoces aux thèmes à venir, ce qui déplace l'occurrence initiale d'un terme lié au thème avant sa position attendue. Néanmoins, l'hypothèse s'avère utile en pratique.

Il y a, ici aussi, plusieurs façons d'implémenter une telle mesure. Par exemple, la version proposée par Hearst (1997, p. 50) calcule le taux moyen par token dans une fenêtre de texte centrée sur la frontière candidate. Nous procédons autrement, pour deux raisons. D'abord, le taux supposé d'introduction de vocabulaire est plus élevé dans le texte qui suit une frontière que dans le texte qui précède ; puisque le domaine du calcul ne fait pas la différence entre les deux, on passe là à côté d'une source d'information potentiellement utile. Ensuite, à moins qu'un ajustement additionnel ne soit appliqué, ce calcul ne tient pas compte du déclin naturel dans le taux d'introduction de vocabulaire observé dans tout texte.

Ainsi, notre système compare le compteur de nouveau vocabulaire dans deux blocs de longueur k de chaque côté de la frontière présumée (la différence dans les taux de blocs adjacents est supposée être négligeable). Comme ailleurs, la contribution de chaque terme est sa valeur RIDF plus que simplement 1. Soit T_i le terme à la position i :

$$nouveau\ (i) = \begin{cases} RIDF(T_i) & s'il\ n'y\ a\ aucun\ j < i\ tel\ que\ T_j = T_i \\ 0 & autrement \end{cases}$$

$$NV'(m,n) = \sum_{i=m}^{n} nouveau(m)$$

$$NV(i) = NV'(i-k, i-1) - NV'(i, i+k)$$

3.7 Entraînement du système

Tel que mentionné ci-dessus, Indexo-II utilise un logiciel d'apprentissage-machine pour extraire du corpus d'entraînement des généralisations sur les caractéristiques des frontières de segments ; il les applique ensuite à l'analyse du corpus test. Plus précisément, il s'agit du package d'arbres de décision OC1 (« Oblique Classifier ») de Murthy et al. (op. cit.). Celui-ci reçoit des vecteurs numériques qui représentent la cohésion lexicale, les scores des chaînes lexicales et le taux d'introduction du vocabulaire des frontières potentielles de segments ; chaque vecteur est annoté à l'aide de sa classe réelle (« frontière » ou « non-frontière »). Un arbre de décision est ensuite produit qui permet d'annoter de nouveaux textes non annotés.

La procédure d'entraînement pour chacun des corpus a procédé comme suit : le texte d'entraînement a été divisé en deux parties, l'une utilisée dans le développement du système et l'autre dans l'entraînement du modèle. Les paramètres du système (c'est-à-dire, les choix variés esquissés ci-dessus pour la pondération des termes, la taille du contexte, etc.) ont été fixés en utilisant les fonctionnalités de validation croisée du package OC1. La partie entraînement du modèle du corpus test a ensuite été analysée avec les paramètres ainsi fixés ; les résultats à cette étape ont servi de données d'entraînement pour l'arbre de décision, qui a alors été appliqué au corpus test.

Il convient de noter que, dans sa configuration la plus exacte, le package OC1 génère des arbres de décision qui se présentent sous la forme de « boîtes noires », dans le sens où on ne

peut extraire, à partir de l'inspection de leur structure, une information utile que l'on pourrait interpréter comme des traits ou des propriétés des données d'entraînement.

4 Contrastes entre diverses approches à la segmentation

En fait, le type d'approche à la segmentation est influencé à la base par l'application visée. Nous discutons, dans les paragraphes qui suivent, de deux tâches différentes : d'abord, celle qui a orienté nos travaux initiaux et qui a trait à l'indexation de monographies (Da Sylva, 2004a, 2004b ; Da Sylva et Doll 2005a, 2005b) ; ensuite, celle qui a dicté les exigences pour la présente implémentation. Les concessions que nous avons faites pour ces deux tâches ne se trouvent pas dans le calcul des indices lexicaux utilisés, mais dans leur combinaison.

4.1 Segmentation pour l'indexation de monographies

Il est utile de présenter en détail l'application visée, puisque ses caractéristiques viennent grandement influencer le type de segmentation nécessaire.

L'indexation automatique de monographies consiste à développer des systèmes qui construisent pour un texte numérique un index constitué d'entrées structurées comme celles, fictives, présentées ci-dessous, que l'on retrouve habituellement à la fin d'un livre (c'est pourquoi ce type d'indexation est souvent appelée « indexation de livre »).

Fièvre
    voir sous Température
…
Température, 186-189
    du bain, 138, 141, 227
    de la chambre, 118, 121, 178
    fièvre, 180, 184, 186-188, 187
    pendant la grossesse, 38
    prise de la, 187
    refroidissement, 178
    urgence, 38, 174
    voir aussi Thermomètre
…
État fiévreux
    voir Fièvre

Ces entrées font référence à un passage déterminé dans le document où apparaît une discussion importante sur le concept exprimé par l'entrée. Dans le cas d'indexation de textes numériques, des références hypertextuelles mènent directement au passage visé.

D'un point de vue théorique, l'indexation de ce type implique d'identifier les thèmes importants abordés dans la monographie, de délimiter les passages où l'on retrouve une information substantielle sur le sujet, et de structurer les termes de l'index. Cette dernière est une opération sémantique complexe : il s'agit notamment de regrouper les thèmes reliés et de distinguer à l'aide de sous-vedettes les références multiples à une même vedette principale (comme pour l'entrée « Température » dans l'exemple ci-dessus). En effet, un index qui contiendrait une douzaine de références à l'entrée « Température » serait jugé passablement inutile. Également, un index qui n'indiquerait pas les variantes que peut prendre une

expression (comme « État fiévreux » et « Fièvre »), ou qui ne redirigerait pas l'utilisateur vers une sous-vedette complexe où se trouve l'information recherchée (« Fièvre », *voir sous* « Température ») serait considéré comme très pauvre au niveau de l'aide à la recherche. La méthodologie pertinente est décrite dans Mulvany (1994) et dans Waller (1999).

Notons que les travaux en indexation automatique de monographies se font rares. Des travaux initiaux (Artandi, 1963 et Earl, 1970) recensent essentiellement les termes les plus fréquents mentionnés dans un document (en utilisant parfois un vocabulaire contrôlé ou thésaurus pour limiter les termes considérés ou en modifiant les termes extraits – Salton, 1988). Une liste alphabétisée est ensuite construite, mais ce résultat est peu probant : aucun tri n'est fait parmi les occurrences des termes pour identifier les passages réellement informatifs, et aucune structuration des entrées n'est effectuée. Nazarenko et Aït el-Mekki (2005), cependant, proposent une implémentation récente qui s'apparente fortement à nos propres travaux (bien que les deux ait été développés indépendamment).

4.1.1 Exigences de la tâche d'indexation de monographies

Du point de vue pratique, une telle indexation nécessite au minimum les étapes suivantes :

- l'identification des mots et expressions utilisés dans le texte
- une délimitation des passages thématiques
- une sélection des entrées à retenir pour chaque passage
- un certain traitement sémantique des candidats termes retenus afin de produire des entrées structurées

Nous n'aborderons pas du tout ici la première étape, ni la quatrième, bien qu'elles soient primordiales. En particulier, le traitement sémantique effectué est ce qui distinguera ce type d'indexation de la recherche en texte intégral. Au sujet de la sélection des entrées, mentionnons simplement que cette sélection déterminera la richesse (ou non) de l'index ; des entrées bien choisies, en nombre juste suffisant, représenteront adéquatement le passage. Il y a ici un lien important avec la segmentation, puisque c'est cette dernière qui définit les passages dans lesquels la sélection de termes s'effectuera. Il importe que ces passages soient bien délimités au niveau thématique ; également, la granularité des passages a une incidence importante sur l'utilité de l'index. Nous nous concentrerons donc sur l'étape de segmentation. Celle-ci est contrainte par les exigences déjà évoquées, que nous discutons plus en détails ci-dessous.

4.1.2 Variabilité de la segmentation

Les travaux antérieurs ont, pour la plupart, pris pour acquis que la segmentation visée pour un texte est unique. Hearst (1997) a soumis un texte à un panel de juges humains et en a généralisé la segmentation « correcte » à partir de leurs jugements individuels, de manière à arriver à un consensus. Cette « bonne réponse » (« *gold standard* ») a été utilisée pour évaluer les performances de l'algorithme. Pour notre part, au contraire, nous faisons l'hypothèse que plusieurs segmentations différentes peuvent être valides, selon les besoins. Par exemple, une segmentation fine permettra plus aisément de faire un résumé détaillé, une plus grossière, un résumé plus général. Le même raisonnement tient pour la production d'un index, dont la taille peut être très variable ; elle peut être déterminée par une limitation en nombre de pages ou en termes de la taille d'un écran d'ordinateur. Il est donc souhaitable de

pouvoir produire un index de taille variable. C'est la segmentation qui représente la contrainte la plus forte sur la taille de l'index. En effet, il faut éviter de segmenter en un très grand nombre de segments (par exemple 100) si la taille de notre index ne doit pas dépasser un seuil beaucoup plus petit (par exemple, 30 entrées), puisqu'alors on doit ou bien combiner des segments ou bien en mettre de côté. Dans nos expérimentations antérieures, il s'est avéré qu'une segmentation à 10% semble donner de bons résultats pour l'indexation (mais n'est pas une valeur fixe : elle est plutôt paramétrable).

La variabilité est atteinte en changeant le nombre de vallées incluses dans la segmentation ; en d'autres termes, c'est le seuil minimal du poids des frontières candidates qui est appelé à varier. La figure 2 ci-dessous illustre le graphe des scores (limités à la cohésion lexicale) pour un texte exemple (Pierre, 2000). On y voit 16 vallées (minimas locaux). Pour deux d'entre elles, le score est à zéro ; elles seront retenues, en ordre, de gauche à droite. Il y a 16 frontières de segments possibles.

Un certain nombre de conséquences découlent de cette méthode. D'abord, plus le seuil est bas, plus le changement thématique devrait être important ; les seuils plus élevés devraient correspondre à des changements thématiques de plus en plus fins. Ensuite, dans une segmentation en $n+1$ segments, $n-1$ segments sont identiques à ceux de la segmentation en n segments ; l'autre segment a été coupé en deux moitiés (peut-être inégales). Ainsi, cette méthode est monotone. Enfin, il existe une limite maximale au nombre de frontières qui peuvent être détectées ; au seuil maximal, lorsque toutes les vallées ont été retenues pour la segmentation, aucune autre frontière ne peut être postulée. Ainsi, le graphique des scores détermine une limite supérieure à la variabilité de la segmentation définie de cette façon.

On peut se baser sur d'autres critères pour implémenter la variabilité, notamment les chaînes lexicales, le taux d'introduction du vocabulaire, etc. Ces possibilités n'ont pas encore été explorées dans nos travaux.

4.1.3 Proportionnalité de la segmentation

Pour segmenter un texte dans le but de l'indexer, il importe de tenir compte de la longueur respective des segments. En effet, si une portion de texte est beaucoup plus grande que les autres, alors la contribution sémantique des petits segments pourra être surreprésentée dans l'index. Ceci suppose que chaque segment fournit un nombre égal de termes d'indexation, ce qui peut ne pas être le cas : les contributions peuvent être proportionnelles à la longueur du segment. Mais même dans ce cas, il n'est pas souhaitable d'avoir des segments trop disproportionnés : il sera plus difficile de faire des choix quant aux termes à retenir dans l'index pour les segments très longs, et les références de l'index seront moins utiles que pour les petits segments, puisque moins ciblées.

De plus, la proportionnalité peut être vue comme un paramètre qui vient contrebalancer la variabilité : la proportionnalité peut s'ajouter au score des frontières pour contraindre les choix parmi celles-ci. Illustrons : les frontières potentielles à la figure 2, liées uniquement au score des frontières, sont distribuées de manière non uniforme dans le texte. Les vallées les plus profondes ne sont pas à distance plus ou moins égale des autres. Si on retient un nombre de frontières uniquement sur le score, la segmentation résultante contiendra des segments de longueurs très inégales. Pour assurer une certaine proportionnalité de la segmentation, on voudra s'approcher des coupures purement proportionnelles (déterminées par la longueur du texte) tout en tenant compte du score des phrases. C'est ainsi que dans Indexo-I, l'algorithme de segmentation combine le score de cohésion lexicale des phrases avec les coupures

purement proportionnelles, à l'aide d'un algorithme vorace, pour produire une segmentation qui tend vers la proportionnalité. L'algorithme est « proportionnel et lexical », d'où son nom PEL. Le nombre de frontières finalement retenu pour la segmentation est déterminé par le ratio de segmentation (le paramètre variable évoqué ci-dessus). À partir d'un même ensemble de scores pour les frontières candidates, différentes frontières finales seront proposées par l'algorithme PEL pour différents ratios de segmentation.

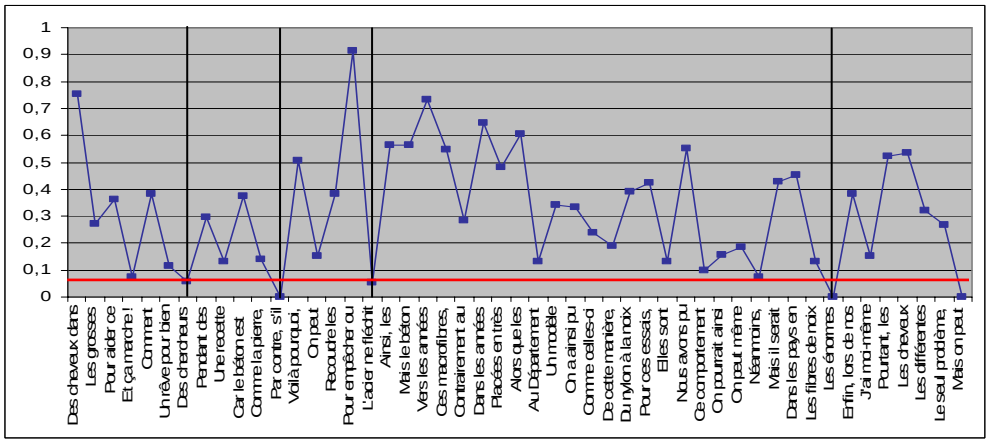

FIGURE 2 – Segmentation basée sur le score de cohésion (4 frontières), seuil≅ 0.06

La figure 3 illustre, pour le texte de Pierre (2000), l'ajustement des frontières à l'aide de PEL (les coupures exactement proportionnelles sont indiquées en lignes pointillées). Avec cet algorithme, une frontière peut être retenue même pour un score de cohésion supérieur au seuil fixé. Les graphiques des deux figures ont été produits par notre premier prototype, Indexo-I ; la mesure de similarité d'une phrase avec la suite est basée essentiellement sur la cohésion mais diffère considérablement de l'algorithme présenté pour la tâche de DEFT'06.

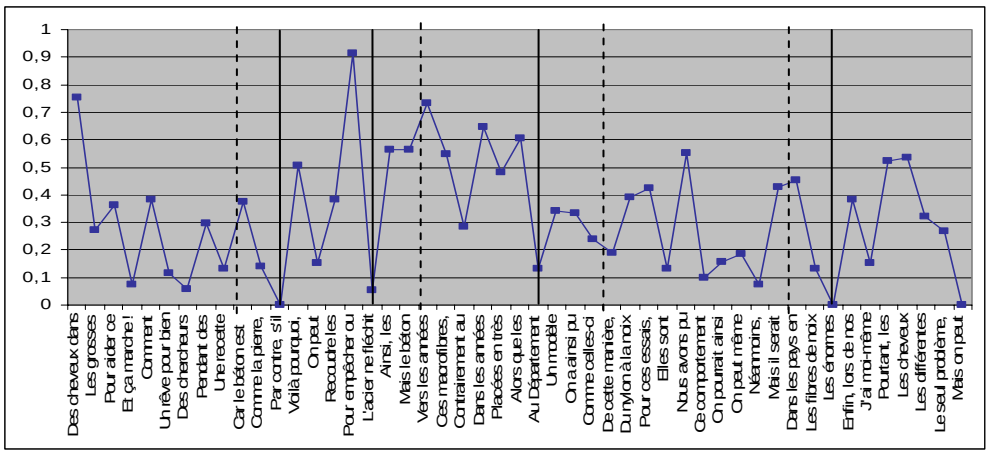

FIGURE 3 – Segmentation en tenant compte de la proportionnalité (4 frontières)

On peut objecter que la proportionnalité de la segmentation n'est pas un but souhaitable, puisqu'assurément certaines sections du texte sont plus complexes et devraient constituer des segments plus importants. Nous répondons que, dans le but de l'indexation, des segments longs et complexes devraient être divisés afin d'être indexés plus finement. Il est vrai que l'introduction et la conclusion d'un texte sont souvent plutôt légères d'un point de vue informationnel, et moins longues, et qu'elles devraient représenter des segments plus courts. Il ne serait pas très difficile d'ajouter cette contrainte à notre algorithme, tout en maintenant la proportionnalité entre les deux. Cependant, pour des raisons pratiques, la construction d'un index bénéficie ultimement d'une segmentation dont les segments ont une taille uniforme.

Nous avons aussi fait des expériences en segmentation automatique de documents XML (ceux qui respectent la DTD DOCBOOK –Marcoux et al., 2005). La prise en compte de la structure explicite est un atout important dans la segmentation de documents ; nos travaux sur cette question se poursuivront sous peu.

4.2 Segmentation pour la tâche de DEFT'06

Pour DEFT'06, il fallait reproduire une segmentation a priori, qui consistait, finalement, à retrouver les positions des titres et sous-titres qui avaient été retirés d'un texte. L'objectif de proportionnalité n'avait aucun sens. La segmentation visée était unique (même si trois « essais » étaient permis pour tenter de la reproduire), et donc la variabilité non plus n'était pas un objectif.

Dans l'optique de reproduire une telle segmentation, nous avons remarqué que la cohésion lexicale accuse certaines lacunes parfois insurmontables. Notamment, dans un texte suivi (comme le corpus scientifique, constitué d'un livre), il est coutumier pour l'auteur d'annoncer, avant une nouvelle section, la teneur de cette section. Ainsi certains mots appartenant véritablement à la thématique d'une section future sont introduits à la fin de la section précédente. Cette introduction précoce de vocabulaire fait invariablement détraquer le score de cohésion. Cette particularité est absente du corpus « Lois ».

Par ailleurs, pour capter certaines dépendances sémantiques d'une phrase à l'autre, il serait parfois nécessaire d'avoir recours à un thésaurus spécialisé. Dans le corpus scientifique, les références aux animaux utilisent un mélange de noms d'espèces spécifiques et génériques (oiseau, palmipède, etc.). Nous avons choisi de ne pas développer de thésaurus ad hoc, et ainsi ces relations échappaient à notre calcul de similarité simple. Également, certaines reprises thématiques évidentes se font à l'aide de paraphrases (notamment quand il s'agit de quantités) qu'il est difficile de capter.

Il est évident que les trois corpus ont des particularités différentes. Pour cette tâche, nous avons préféré utiliser un algorithme adapté à chaque corpus, par le biais de l'arbre de décision ; pour un corpus quelconque, un nouvel entraînement serait sans doute nécessaire.

Nous avons expérimenté avec différentes façons de calculer le score de cohésion lexicale. Notamment, nous avons comparé les performances relatives de la lemmatisation et de la radicalisation des mots. Il s'est avéré que la radicalisation donne de meilleurs résultats. Notons que pour l'indexation automatique de monographies, il n'est pas clair que la radicalisation soit préférable, puisque l'analyse sémantique requise pour produire les entrées d'index est plus exigeante que celle utilisée pour la segmentation.

Une autre différence notable entre l'implémentation Indexo-I et Indexo-II est l'unité de comparaison : Indexo-I compare entre elles des phrases entières ou alors que Indexo-II (comme Hearst, 1997) utilise des fenêtres de mots.

5 Évaluation

Nous présentons d'abord les résultats numériques obtenus pour l'évaluation de notre algorithme, puis nous commentons certaines des difficultés rencontrées.

5.1 Résultats obtenus

Les tableaux 2 à 4 font état des résultats obtenus par notre algorithme, selon les données fournies par les organisateurs. Ainsi, la précision globale atteinte est de 58,04%, le rappel global, de 25,59% (f-score global de 33,69%). Dans le cas de la précision, nous sommes bien au-dessus de la moyenne qui est à 22,06%. Par contre, nous sommes en-dessous du rappel global, à 34,48%. La meilleure cote obtenue l'est pour le corpus « Lois » : notre précision moyenne globale de 77,66% dépasse largement la moyenne (20,6%), alors que le rappel de 47,83% dépasse aussi le rappel moyen des équipes (à 26,65%). Pour les deux autres corpus, notre f-score se compare presque toujours à la moyenne des équipes, ce qui est obtenu généralement à l'aide de notre précision élevée. La pire cote est celle du rappel dans le corpus « Discours » : 11,83%, comparativement à 41,14% pour la moyenne des équipes.

		Corpus Discours		Corpus Lois		Corpus Scientifique	
		Nos résultats	Moyenne des équipes	Nos résultats	Moyenne des équipes	Nos résultats	Moyenne des équipes
Strict	Précision	0,592901	0,187377311	0,756401	0,170131223	0,1684210	0,090204811
	Rappel	0,107228	0,244465578	0,470143	0,210563079	0,0941176	0,202614383
	f-score	0,181611	0,1814025	0,579868	0,170656786	0,1207550	0,115022639
Souple-1	Précision	0,669790	0,296970889	0,756654	0,199535394	0,3229170	0,174697167
	Rappel	0,118957	0,435176389	0,470301	0,258372356	0,1823530	0,388235278
	f-score	0,202032	0,302952944	0,580062	0,204803222	0,2330830	0,2196155
Souple-2	Précision	0,726866	0,384874222	0,817022	0,248393733	0,4123710	0,233323889
	Rappel	0,128864	0,554639889	0,494564	0,330453622	0,2352940	0,478758167
	f-score	0,218916	0,394533111	0,616155	0,258744294	0,2996250	0,287356778

TABLEAU 2 – Résultats des tests

	Précision		Rappel		f-score	
	Nos résultats	Moyenne des équipes	Nos résultats	Moyenne des équipes	Nos résultats	Moyenne des équipes
Strict	0,5059077	0,149237782	0,2238295	0,219214347	0,2940780	0,155693975
Souple-1	0,5831203	0,223734483	0,2572037	0,360594674	0,3383923	0,242457222
Souple-2	0,6520863	0,288863948	0,2862407	0,454617226	0,3782320	0,313544728
Globale	0,5803714	0,220612071	0,2557580	0,344808749	0,3369008	0,237231975

TABLEAU 3 – Moyennes par mesure

À cause d'un malentendu au niveau de la forme des soumissions, nous avons soumis une exécution par corpus aux tests, plutôt que trois exécutions contenant chacun des trois corpus. Or, nous aurions peut-être gagné à en soumettre plus d'une. Mais en fait, notre algorithme n'est pas lié à un seuil quelconque ; les paramètres du système ont été fixés à des valeurs optimales durant la phase d'entraînement, ce qui a été jugé suffisant pour effectuer les tests.

		Nos résultats	Moyenne des équipes	Écart-type des équipes
Corpus discours	précision	0,663185667	0,289740807	0,13567677
	rappel	0,118349667	0,411427285	0,21174024
	f-score	0,200853000	0,292962852	0,09353902
Corpus lois	précision	0,776692333	0,206020117	0,22490196
	rappel	0,478336000	0,266463019	0,24139552
	f-score	0,592028333	0,211401434	0,19611511
Corpus scientifique	précision	0,301236333	0,166075289	0,10041194
	rappel	0,170588200	0,356535943	0,25667494
	f-score	0,217821000	0,207331639	0,12196696

TABLEAU 4 – Moyennes et écart-types par corpus

5.2 Caractéristiques du corpus et performance du système

Une hypothèse importante sous-tend ce travail, soit que les patrons de distribution lexicale représentent un bon indicateur de la structure thématique. Il y a lieu d'examiner quels facteurs peuvent influencer l'exactitude de cette hypothèse.

Supposons une situation où chaque mot d'un texte (ou chaque mot « plein », du moins) apparaît uniquement dans une unité thématique. Dans ce cas, la cohésion lexicale et les scores des chaînes lexicales seraient de zéro aux frontières d'unités et plus élevées ailleurs, et l'on pourrait s'attendre à ce qu'un système comme Indexo-II atteigne de très bonnes performances. Cependant, les textes réels diffèrent de cette situation extrême, de plus d'une façon. Dans certains genres textuels, on considère qu'un bon style d'écriture est caractérisé par l'introduction de nouvelles notions avant que celles-ci soient discutées en détail ; les articles techniques et savants contiennent typiquement de longs passages ayant cette propriété. Également, des articles, sections ou chapitres se terminent fréquemment par un résumé du contenu précédent qui récapitule les thèmes qui y ont été présentés. De manière plus générale, on remarque souvent dans les textes descriptifs ou didactiques la technique suivante : « dites-leur ce que vous allez raconter, racontez-le, puis dites-leur ce que vous leur avez raconté ». Il devient alors plus difficile de faire coïncider la cohésion thématique et la distribution du vocabulaire. De plus, il arrive très souvent qu'un thème soit repris en détail plus loin dans le texte ; par conséquent, le seul taux d'introduction du vocabulaire n'est pas un indice fiable.

Pour les textes utilisés dans l'évaluation DEFT'06, il n'est pas surprenant que la performance d'Indexo-II sur le texte de lois soit supérieure ; ceci est un bon exemple d'un genre où les techniques rhétoriques évoquées ci-dessus sont rares. De plus, il semble plausible que les traits textuels supprimés des textes réels pour produire les corpus de DEFT'06 correspondent plus étroitement avec les changements de thématique dans ce type de texte, que dans les genres relativement informels du manuel scientifique ou du discours politique.

5.3 Difficulté de la tâche d'évaluation

Il y a lieu de procéder à davantage d'évaluations des résultats de la segmentation. La difficulté d'obtenir des jugements humains pour cette tâche (ainsi que pour la thématisation des segments, qui est une question que nous n'avons pas abordée ici) est notoire (certains problèmes

inhérents sont discutés dans Passonneau et Litman, 1993). La présentation de l'indexation de monographies présentée ci-dessus fait de plus ressortir le fait que la segmentation étant rarement une finalité, il est nécessaire de tenir compte de l'application visée afin de définir des critères requis de la segmentation, et des métriques d'évaluation appropriées.

Dans nos propres expériences d'évaluation pour Indexo-I, nous nous sommes servis d'une notion de distance d'édition (similaire à celle de Levenshtein, 1966) entre les frontières visées et les frontières trouvées par l'algorithme de segmentation. Cette distance d'édition est légèrement différente de la mesure « souple » utilisée dans l'évaluation des résultats de DEFT'06, dans la mesure où elle permet un éloignement quelconque entre les frontières visées et les frontières obtenues. Dans nos travaux de segmentation variable, nous avions de plus incorporé la notion de « surplus acceptables » ou « déficits acceptables » : en effet, pour une segmentation (humaine) en n segments, si notre algorithme de segmentation produisait $n+m$ segments (sur demande), il avait droit à m frontières en plus, sans pénalité aucune (ou en moins, pour une segmentation en $n-m$ segments).

Bien sûr, la vraie notion à capter par ces algorithmes automatiques est celle de cohésion thématique, et non purement lexicale. Les ajouts nécessaires à l'algorithme de base décrit ici incluent entre autres l'utilisation d'un thésaurus ainsi que la reconnaissance (et même la résolution) des anaphores. Dans notre implémentation Indexo-I, certaines anaphores initiales sont prises en compte (mais non résolues) et viennent ainsi augmenter le score de cohésion d'une phrase à l'autre.

6 Conclusion

Nous avons développé un algorithme de segmentation automatique qui utilise plusieurs informations relatives à la distribution des mots dans le texte pour proposer des frontières candidates. Les résultats de précision que nous avons obtenus sont très encourageants, étant largement au-delà de la moyenne des équipes participant à DEFT'06. Le rappel cependant laisse passablement à désirer. Finalement, pour le corpus « Lois » spécifiquement, nos résultats sont bien supérieurs à la moyenne.

Ces résultats nous laissent croire que la combinaison des critères retenus représente une voie prometteuse. Il demeure cependant un nombre d'améliorations à apporter, notamment l'inclusion de connaissances thésaurales et le traitement d'anaphores.

Références

Artandi S. (1963), *Book indexing by computer*, New Brunswick, N.J.: S.S. Artandi.

Barzilay R. et Elhadad M. (1997), Using lexical chains for text summarization, in *Intelligent Scalable Text Summarization. Proceedings of a Workshop*, pp 10-17.

Church K.W. (1995), One Term or Two?, in *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp 310-318.

Da Sylva L. et Doll F. (2005a), A Document Browsing Tool: Using Lexical Classes to Convey Information, in Lapalme G. et Kégl B. (reds), *Advances in Artificial Intelligence: 18th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2005 (Proceedings)*, New York : Springer-Verlag, pp 307-318.

Da Sylva L. et Doll F. (2005b), Information Architecture for Document Description: Semantic Thematization of Text Segments, in Tochtermann K. et Maurer H. (réds), *Proceedings of I-KNOW '05. 5th International Conference on Knowledge Management*, Graz, Autriche, 29 juin – 1er juillet, pp 612-620.

Da Sylva L. (2004a), Relations sémantiques pour l'indexation automatique. Définition d'objectifs pour la détection automatique, *Document numérique*, Numéro spécial « Fouille de textes et organisation de documents », 8, 3, pp 135-155.

Da Sylva L. (2004b), A Document Browsing Tool Based on Book Indexes, in *Proceedings of Computational Linguistics in the North East*, Concordia University, Montréal, pp 45-52.

Earl, L.L. (1970), Experiments in automatic extraction and indexing, *Information Storage and Retrieval*, 6, pp 313-334.

Halliday M. et Hasan R. (1976), *Cohesion in English*, London: Longman.

Levenshtein V.I. (1966), Binary codes capable of correcting deletions, insertions, and reversals, *Doklady Akademii Nauk SSSR*, 163(4), pp 845-848, 1965 (en russe). Traduction anglaise dans *Soviet Physics Doklady*, 10(8), pp 707-710, 1966. [voir aussi http://en.wikipedia.org/wiki/Edit_distance]

Manning C. et Schütze H. (1999), *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, Mass.

Marcoux Y., Da Sylva L. et Doll, F. (2005), *Indexation automatique de documents XML*, 73e congrès de l'ACFAS, section Sciences de l'information, Université du Québec à Chicoutimi, 9 mai 2005.

Morris J. et Hirst G. (1991), Lexical cohesion computed by thesaural relations as an indicator of the structure of text, *Computational Linguistics* 17(1), pp 21-43.

Mulvany N. (1994), *Indexing Books*, Chicago : University of Chicago Press.

Murthy S.K., Kasif K. et Salzberg S. (1994), A System for Induction of Oblique Decision Trees, *Journal of Artificial Intelligence Research* 2, pp 1-32.

Nazarenko A. et Aït el-Mekki T. (2005), Building back-of-the-book indexes, *Terminology*, 11(1), pp 199-224.

Passonneau R.J. et Litman D.J. (1993), Intention-based segmentation: Human reliability and correlation with linguistic cues, *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pp 148–155.

Patry R., Ménard N. et Ponzio J. (1989), La Question des nombres dans l'analyse de la cohésion textuelle: une innovation méthodologique, *Revue québécoise de linguistique théorique et appliquée*, 8(3-4), pp 107-126.

Salton, G. (1988), Syntactic Approaches to Automatic Book Indexing, in *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, 7-10 juin 1988, State University of New York at Buffalo, Buffalo, New York. Morristown, N.J. : Association for Computational Linguistics, pp 204-210.

Waller, S. (1999), *L'analyse documentaire. Une approche méthodologique*, Paris : ADBS Éditions.

Yaakov Y. (1997), Segmentation of expository texts by hierarchical agglomerative clustering, in *Proceedings of RANLP'97*, Bulgaria.

Summary

This paper presents the work of the GRDS team in the context of the DEFT'06 competition. An algorithm for automatic text segmentation is described, which uses a number of information sources relative to lexical distribution in the text to propose candidate segment boundaries: lexical cohesion, lexical chains, and the rate of vocabulary introduction. A decision tree is built for each of the training corpora. Precision results average 58.04% (well above the average obtained by participating teams) whereas recall is 25.59%. For one of the corpora, overall results are much higher than the teams' average (f-score of 59.20%, compared to 21.14% on average). The algorithm is compared to another similar one, used in an automatic *back-of-the-book* indexing system, in order to illustrate the impact of the segmentation's final use on the segmentation task itself.