

REG :

Un algorithme glouton appliqué au résumé automatique de texte. Une approche exploratoire

Juan-Manuel Torres-Moreno ^{1,2}, Javier Ramírez Rodríguez ³

¹ UAPV-LIA – F-84911 Avignon cedex – France

² Dép. Génie Informatique-École Polytechnique de Montréal, H3C3A7 (Québec) Canada

³ UAM-Azcapotzalco - Av. San Pablo 180, CP 02200 Mexico DF - Mexique

Résumé

Dans cet article nous présentons REG, une approche de graphes pour étudier un problème fondamental du Traitement Automatique de la Langue Naturelle : le résumé automatique de documents. L'algorithme modélise un document comme un graphe où l'on déduit la pondération des phrases. Nous avons appliqué cette approche à la génération de résumés par extraction avec des résultats très encourageants en trois langues.

Abstract

In this paper we introduce a graph approach to the fundamental natural language processing problem: automatic text summary. The algorithm models a document like a graph to obtain weighted sentences. We applied this approach to automatic text summary by extraction with promising results.

Keywords : automatic summary, optimisation methods, vectorial model, algorithm of graphs

1. Introduction

Le résumé automatique de documents est une des méthodes de la fouille de textes qui permet de compresser un document tout en conservant son informativité. Il s'agit d'une problématique importante du Traitement de la Langue Naturelle Écrite (TALNE). Résumer consiste à condenser l'information la plus importante provenant d'un ou de plusieurs documents, afin d'en produire une version abrégée de son contenu (Inderjeet and Mark, 1999). Les gros titres des nouvelles, les bandes-annonces et les synopsis sont quelques exemples de résumés que nous utilisons couramment. De manière générale, les personnes sont des résumeurs extrêmement performants. Les approches par extraction de phrases combinant des algorithmes numériques/statistiques ont montré leur pertinence dans cette tâche difficile. En se basant sur les études du comportement des résumeurs professionnels et notamment sur les travaux de (Kintsch and van Dijk, 1978 ; Van Dijk, 1979), les chercheurs ont essayé d'imiter le processus cognitif de création d'un résumé. Les premiers travaux portant sur le résumé automatique de textes datent de la fin des années 50 (Luhn, 1958). Luhn décrit une technique simple, spécifique aux articles scientifiques qui utilise la distribution des fréquences de mots dans le document pour pondérer les phrases. Luhn était déjà motivé par la problématique de surcharge d'information.

Il décrit quelques uns des avantages qu'ont les résumés produits de manière automatique par rapport aux résumés manuels : coût de production très réduit, non assujetti aux problèmes de subjectivité et de variabilité observés sur les résumés professionnels. L'idée d'utiliser des techniques statistiques pour la production automatique de résumés a eu un impact considérable, la grande majorité des systèmes d'aujourd'hui étant basés sur ces mêmes idées. Par la suite, (Edmundson, 1969) a étendu les travaux de Luhn en tenant compte de la position des phrases, de la présence des mots provenant de la structure du document (par exemple les titres, sous-titres, etc.) et de la présence de mots indices (*e.g.* « significant », « impossible », « hardly », etc.). Les recherches menées par (Pollock and Zamora, 1975) au sein du *Chemical Abstracts Service* (CAS) dans la production de résumés à partir d'articles scientifiques de chimie ont permis de valider la viabilité des approches d'extraction automatique de phrases. Un nettoyage des phrases reposant sur des opérations d'élimination fut pour la première fois introduit. Les phrases commençant par exemple par « in » (par exemple « in conclusion ») ou finissant par « that » seront éliminées du résumé. Afin que les résumés satisfassent les standards imposés par le CAS, une normalisation du vocabulaire est effectuée. Elle inclut le remplacement des mots/phrases par leurs abréviations, une standardisation des variantes orthographiques. Ces travaux ont posé les bases du résumé automatique de textes. Une méthodologie de production des résumés émerge de leur analyse : i) Prétraitement, ii) Identification des phrases saillantes dans le document source, iii) Construction du résumé par concaténation des phrases extraites et iv) Traitement de surface des phrases. Le travail présenté dans cet article porte sur la conception d'un système de résumé automatique générique. Ce système utilise des méthodes de graphes afin de repérer dans le document les phrases les plus importantes. Notre volonté de n'utiliser que des traitements statistiques est motivée par le fait que le système doit être le plus indépendant possible de la langue. La méthode que nous proposons repose sur un prétraitement spécifique des documents et sur une fonction de pondération des phrases utilisant optimisation dans un graphe. Dans un premier temps, nous allons décrire en détail les différentes méthodes utilisées par notre système afin de préparer les documents et de donner un *ranking* à chaque phrase. Ensuite, nous nous intéresserons à l'évaluation des résumés produits et nous discuterons de la validité de ces résultats avant de conclure et de donner quelques perspectives.

2. Algorithmes de résumé à base de graphes

Mihalcea (2004), Erkan et Radev (2004) considèrent le résumé par extraction comme une identification des segments les plus *prestigieux* d'un graphe. Les algorithmes de classement basés sur les graphes tel que *PageRank* (Brin and Page, 1998) ont été utilisés avec succès dans les réseaux sociaux, l'analyse du nombre de citations ou l'étude de la structure du Web. Ces algorithmes peuvent être vus comme les éléments clés du paradigme amorcé dans le domaine de la recherche sur Internet, à savoir le classement des pages Web par l'analyse de leurs positions dans le réseau et non pas leurs contenus. En d'autres termes, ces algorithmes permettent de décider de l'importance du sommet d'un graphe en se basant non pas sur l'analyse locale du sommet lui même, mais sur l'information globale issue de l'analyse récursive du graphe complet. Appliqué au résumé automatique, cela signifie que le document est représenté par un graphe d'unités textuelles (phrases) liées entre elles par des relations issues de calculs de similarité. Les phrases sont ensuite sélectionnées selon des critères de *centralité* dans le graphe puis assemblées pour produire des extraits. Les résultats rapportés montrent que les performances des approches à base de graphes sont au niveau des meilleurs systèmes actuels (Mihalcea, 2005) mais ne portent que sur des documents en anglais et en portugais. Il est important de noter que les méthodes de classement sont entièrement dépendantes de la bonne

construction du graphe censé représenter le document. Puisque ce graphe est généré à partir de mesures de similarités inter-phrases, l'impact que peut avoir le choix de la méthode de calcul est à considérer.

Dans leurs travaux, (Mihalcea, 2004 ; Erkan and Radev, 2004) utilisent le modèle en sac-de-mots pour représenter chaque phrase comme un vecteur à N dimensions, où N est le nombre total de mots différents du regroupement et chaque composante du vecteur représente un poids $tf \times idf$. Les valeurs de similarité entre phrases sont ensuite obtenues par un calcul du cosinus entre leurs représentations vectorielles. Le point faible de cette mesure, et plus généralement de toutes les mesures utilisant les mots comme unités, est qu'elles sont tributaires du vocabulaire. Dans une optique d'indépendance de la langue, les pré-traitements qui sont appliqués aux segments se doivent d'être minimaux. C'est malheureusement dans cette configuration que les performances de la mesure cosinus chutent car elle ne permet en aucun cas de mettre en relation des mots qui morphologiquement peuvent être très proches. Une solution intéressante combine les mesures de similarité et celles basées sur les caractères. (Boudin et al., 2008) proposent une mesure dérivée d'un calcul de similarité entre chaînes de caractères originellement employé pour la détection d'entités redondantes (*Record Linkage*). Cette mesure permet de créer des relations entre deux segments même s'ils ne partagent aucun mot, mais en contiennent des morphologiquement proches. Une seconde question est donc de savoir si la construction du graphe du document à partir de mesures mixtes (mots et caractères) permet d'améliorer l'extraction de segments. (Boudin and Torres, 2009) ont montré que cela est possible. Cependant, nous voulions une solution avec un algorithme à base de graphes encore plus simple. Nous posons le problème du résumé automatique de texte par extraction comme un problème d'optimisation. Ainsi, un texte est représenté comme un graphe non dirigé qui peut être assimilé comme un problème de coloration ou à une des variantes de celui du voyageur du commerce. Le problème ainsi posé est de l'ordre de $P!$, étant P le nombre de phrases d'un document. Cela fait de cette tâche un problème NP -complet. Alors, nous nous sommes tournés vers les approches gloutonnes. Nous avons développé un algorithme optimal de visite des m sommets, étant m fixé par l'utilisateur. L'algorithme REG (REsumeur à base de Graphes) réalise l'extraction des m phrases les plus pertinentes qui constitueront le résumé par extraction d'un texte.

3. REG : Un algorithme résumeur glouton

La méthode REG consiste en deux grandes phases : d'abord une représentation adéquate des documents, puis une pondération des phrases. La première est réalisée au moyen d'une représentation vectorielle. La deuxième par un algorithme d'optimisation glouton. La génération du résumé est effectuée par concaténation des phrases pertinentes, pondérées dans l'étape d'optimisation.

3.1. Pré-traitement et représentation vectorielle

Les documents sont pré-traités avec des algorithmes classiques de filtrage de mots fonctionnels (nous avons effectué le filtrage de chiffres et exploité des anti-dictionnaires), de normalisation et de lemmatisation (Porter, 1980 ; Mani and Maybury, 1999) afin de réduire la dimensionnalité.

Une représentation en sac de mots produit une matrice $S_{[P \times N]}$ de fréquences/absences composée de $\mu = 1, \dots, P$ phrases (lignes) ; $\sigma_{\mu} = \{s_{\mu,1}, \dots, s_{\mu,i}, \dots, s_{\mu,N}\}$ et un vocabulaire de $i = 1, \dots, N$ termes (colonnes).

$$S = \begin{pmatrix} S_{1,1} & S_{1,2} & \cdots & S_{1,N} \\ S_{2,1} & S_{2,2} & \cdots & S_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ S_{P,1} & S_{P,2} & \cdots & S_{P,N} \end{pmatrix}; \quad S_{\mu,i} = \begin{cases} TF_i & \text{si le terme } i \text{ existe} \\ 0 & \text{autrement} \end{cases} \quad (1)$$

La présence du mot i est représentée par sa fréquence TF_i (son absence par 0 respectivement), et une phrase σ_μ est donc un vecteur de N occurrences. S est une matrice entière car ses éléments prennent des valeurs fréquentielles absolues.

3.2. Solution gloutonne

A partir du modèle vectoriel de représentation de documents, nous proposons de créer un graphe $G = (S, A)$ où les sommets S représentent les phrases et A l'ensemble d'arêtes. Une arête entre deux sommets est créée si les phrases correspondantes possèdent au moins un mot en commun. On construit une matrice d'adjacence à partir de la matrice $S_{[\text{phrases} \times \text{mots}]}$ comme suit : Si l'élément $S_{i,k} = 1$ de la matrice S (dans la phrase i le mot k est présent), on vérifie dans la colonne k et quand un élément $S_{j,k} = 1$ on met 1 dans la case a_{ij} de la matrice d'adjacence A , ce qui veut dire que les phrases i et j partagent le mot k . Pour afficher les phrases les plus lourdes nous avons trouvé qu'il fallait chercher une variante du problème de l'arbre de poids maximum, où les poids sont sur les sommets, pas sur les arêtes. Nous avons ainsi construit un algorithme inspiré de l'algorithme de Kruskal (Gould, 1988).

L'algorithme proposé fonctionne de la façon suivante :

1. Générer la matrice d'adjacence A qui aura autant des lignes et des colonnes que des phrases considérées, c'est à dire P ;
2. Calculer le poids des sommets, c'est-à-dire la somme d'arêtes entrantes du sommet ;
3. Calculer le degré de chaque sommet : le nombre de mots partagé avec les autres phrases.

La matrice d'adjacence $A_{[P \times P]}$ sera générée à partir de la représentation vectorielle (équation 1). Le calcul est comme suit : parcourir la ligne $i=1 \dots P$, et pour chaque élément a_{ij} égal à 1, descendre par la colonne j pour identifier d'autres phrases qui partagent ce mot. $a_{ij} = 1$ si un mot présent dans la phrase i l'est aussi dans la phrase j ; 0 autrement. La solution que nous avons retenue repose sur un calcul glouton de recherche de chemins.

Algorithm 1 REG: Un REsumeur Glouton

Require: $L, U = V$, Degré de chaque sommet, $T \leftarrow \infty$

Trier les arêtes de G par ordre croissant du poids w

Ajouter à T les éléments de la liste ordonnée comme suit

$T = V(i); i = 1$

1. Si l'arête $(V(i), V(i+1))$ existe, ajouter $V(i+1)$ à T , $V(i+1)$ est dans U
 2. $T = T \cup V(i+1)$
 3. $U = U - V(i+1)$
 4. si non, aller à 3
 5. Faire $i = i + 1$
 6. Si $|T| = L$, arreter, si non aller à 2
 7. rendre la séquence des sommets calculée
-

Figure 1 : Algorithme REG

L’algorithme REG (Figure 1) réalise les étapes suivantes :

1. Choisir le sommet le plus lourd v_0 , et le mettre dans T . Il sera appelé **racine**.
2. La racine sera choisie parmi les nœuds dont le degré est supérieur ou égal à deux.
3. Ajouter à T le voisin de v_0 le plus lourd. Il sera choisi parmi ceux qui ne font pas partie de T .
4. Répéter 2 jusqu’en avoir les k sommets requis.
5. La sortie sera le chemin T .

Pour montrer le fonctionnement de l’algorithme REG, nous allons présenter son application sur deux exemples.

4. Expériences et résultats sur le résumé automatique

Sous l’hypothèse que le poids d’une phrase μ indique son importance dans le document, nous avons appliqué l’algorithme 1 au résumé par extraction de phrases (Mani and Maybury, 1999 ; Radev et al., 2002). Notre méthode est orientée, pour le moment, à la génération de résumés génériques mono-document. L’algorithme REG de résumé automatique comprend trois modules. Le premier réalise la transformation vectorielle du texte avec des processus de filtrage, de lemmatisation/*stemming* et de normalisation. Le second module applique l’algorithme glouton et réalise le calcul de la matrice d’adjacence. Nous obtenons la pondération de la phrase v directement de l’algorithme. Ainsi, les phrases pertinentes seront sélectionnées comme ayant la plus grande pondération. Finalement, le troisième module génère les résumés par affichage et concaténation des phrases pertinentes. Le premier et le dernier module reposent sur le système Cortex (Torres et al., 2002 ; Boudin and Torres, 2007), qui effectue une extraction non supervisée de phrases pertinentes en utilisant plusieurs métriques pilotées par un algorithme de décision.

Nous avons évalué les résumés produits par notre système avec le logiciel ROUGE (Lin, 2004), qui mesure la similarité, suivant plusieurs stratégies, entre un résumé candidat (produit automatiquement) et des résumés de référence (créés par des humains par extraction ou par abstraction). Le texte « Mars » (voir l’Annexe) contient un titre et 11 phrases. Après les processus de pré-traitement et de vectorisation, nous obtenons une matrice S de $P=11$ phrases (commençant en la 0) et de $N=16$ termes :

$$S_{[P=11 \times N=16]}^{Mars} = \begin{matrix} & \text{mot } j & \\ & \downarrow & \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} & \leftarrow \text{phrase } i \end{matrix}$$

Les poids des 11 sommets (représentant les phrases) du graphe correspondant sont : $a_0=2, a_1=5, a_2=2, a_3=7, a_4=3, a_5=1, a_6=1, a_7=6, a_8=4, a_9=4, a_{10}=1$. La matrice A d’adjacence de $P \times P$ phrases est alors :

Le texte « Puces » contient 30 phrases (ce texte est récupérable à l'adresse : <http://lia.univ-avignon.fr/fileadmin/documents/Users/Intranet/chercheurs/torres/corpus.html>). Il est composé de deux thématiques différentes : la première traite les puces électroniques et la deuxième puces biologiques. Ce texte composite a été étudié dans (Torres et al., 2002). On attend des systèmes de résumé sur des textes composites de ce type, qu'ils puissent générer des résumés équilibrés par rapport aux deux thématiques. Les poids des 30 sommets sont : $a_0=a_3=a_6=2$, $a_1=8$, $a_2=5$, $a_4=4$, $a_5=7$, $a_7=5$, $a_8=9$, $a_9=1$, $a_{10}=7$, $a_{11}=a_{12}=5$, $a_{13}=a_{14}=a_{29}=3$, $a_{15}=2$, $a_{16}=10$, $a_{17}=3$, $a_{18}=4$, $a_{19}=1$, $a_{20}=8$, $a_{21}=a_{22}=3$, $a_{23}=1$, $a_{24}=5$, $a_{25}=1$, $a_{26}=a_{27}=2$, $a_{28}=5$. L'application de l'algorithme donne la permutation : 16, 20, 2, 11, 8, 1, 5, 10, 12, 7, 4, 14, 13, 0, 5. Pour un résumé de 8 phrases (25% de la taille en phrases) la suite de sommets obtenue par l'algorithme est : $16 \rightarrow 20 \rightarrow 2 \rightarrow 11 \rightarrow 1 \rightarrow 8 \rightarrow 10 \rightarrow 5$ et le résumé sera généré par la concaténation des phrases 1, 2, 5, 8, 10, 11, 16 et 20.

Nous avons réalisé une batterie de tests différents sur un corpus de documents très hétéroclite (732 phrases, 18.270 mots). Des évaluations de textes en français du domaine ouvert (textes composites et littéraires) ; textes en anglais du domaine encyclopédique ; et textes en espagnol d'un domaine de spécialité. Nous allons décrire en détail les expériences réalisées. Pour l'évaluation des tests en français (textes récupérables sur le site <http://www.lia.univ-avignon.fr>) nous avons choisi le corpus suivant : « Mars », « Puces » et la lettre d'Emile Zola « J'accuse » (<http://fr.wikipedia.org/wiki/J'accuse>). Deux textes de Wikipédia en anglais ont été analysés, « Lewinsky » http://en.wikipedia.org/wiki/Monica_Lewinsky, et « Québec » http://en.wikipedia.org/wiki/Quebec_sovereignty_movement. Enfin, en espagnol nous avons utilisé des textes spécialisés de la revue « Medicina Clínica » http://www.elsevier.es/revistas/ctl_servlet?f=7032&revistaid=2. Pour ce dernier test, un corpus composé de huit textes (environ 400 phrases et 11.000 mots) a été sélectionné. Nous avons évalué les résumés produits par notre système avec ROUGE. Dans le cas des corpus français et anglais, les résumés de référence ont été produits par plusieurs juges de niveau d'études universitaires. Pour le corpus en espagnol, nous avons utilisé les résumés produits par les auteurs comme résumé de référence. Dans Tab. 1 nous présentons le détail des mesures Rouge-2 et SU4 pour le texte « Mars ». Dans cette table on constate que les trois premières places sont *ex-aequo* par REG, Cortex et Enertex (Fernandez et al., 2008).

Rouge	REG	Cortex	Enertex	OTS	Copernic	Word	Random	Leadbase	Pertinence	Swseum
-2	0.8198	0.8198	0.8198	0.5554	0.6620	0.1609	0.3923	0.3262	0.5117	0.0319
SU4	0.8128	0.8128	0.8128	0.5716	0.6729	0.1627	0.3999	0.3271	0.5371	0.0560

Table 1 : Résultats ROUGE pour les différents systèmes. Texte en français : Mars (11 phrases, 241 mots ; résumé à 25%=3 phrases ; 11 résumés de référence)

Nous comparons dans les figg. 3, 4 et 5 les performances de la méthode REG avec celle d'Enertex, de Cortex, de plusieurs systèmes état de l'art et commerciales et de deux *baselines* : une où les phrases ont été choisies au hasard (*Random*) et une autre contenant les premières phrases du document (*Leadbase*). Cette dernière peut être très dure à battre, car les documents ouverts ou de type journalistique concentrent l'information au début du document. Nous constatons que notre méthode est toujours comparable aux systèmes Enertex et Cortex, en se situant parfois au-dessus de ce dernier, réputé par la qualité de ses résumés.

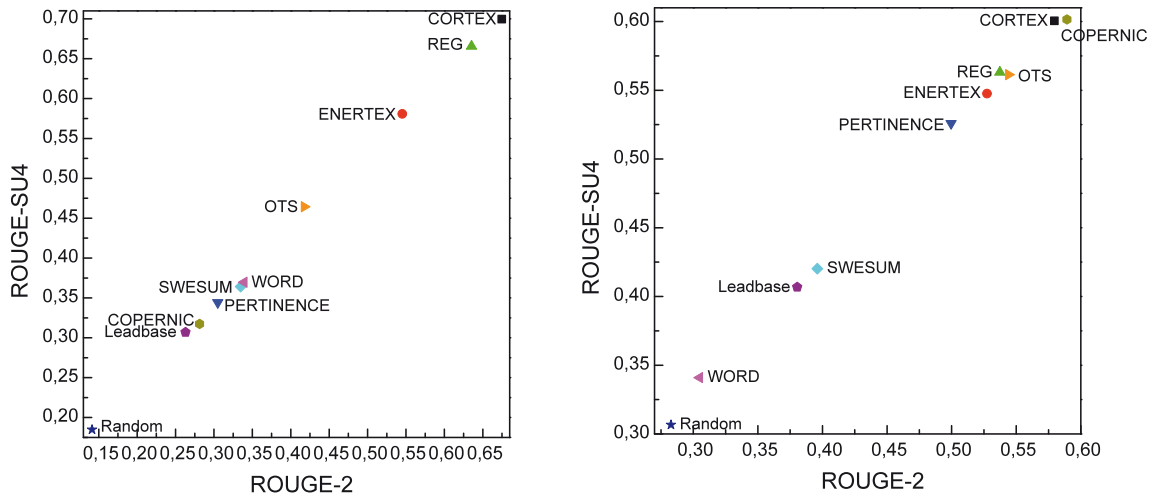


Figure 3 : Rouge-2 et SU4 pour les textes « J'accuse » (à gauche, 206 phrases, 4 936 mots ; résumé à 12% = 25 phrases ; 5 résumés de référence) et « Puces » (à droite, 30 phrases, 607 mots ; résumé à 25%=8 phrases ; 31 résumés de référence)

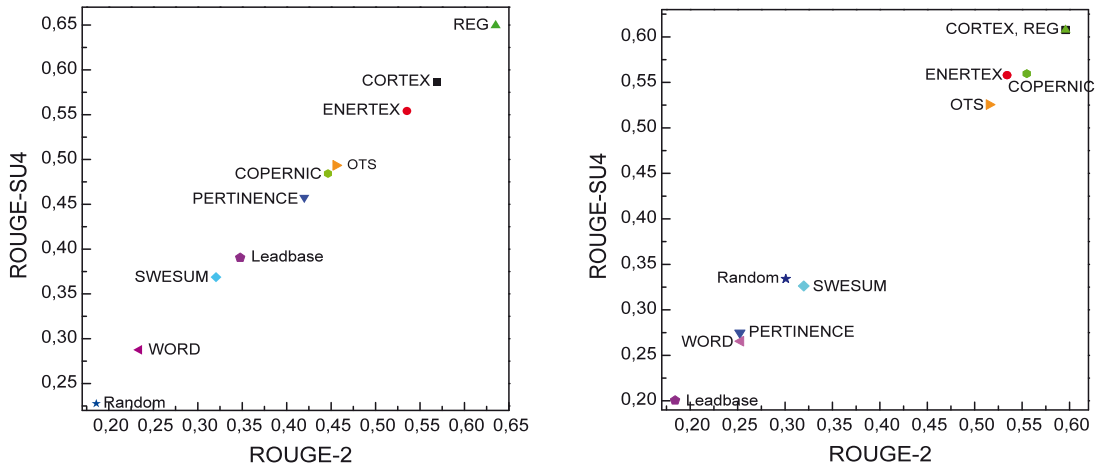


Figure 4 : Tests en anglais. Rouge-2 et SU4 pour les textes « Québec » (à gauche, 44 phrases, 1.184 mots ; résumé à 25%=11 phrases ; 5 résumés de référence) et « Lewinsky » (à droite, 30 phrases, 811 mots ; résumé à 20%=7 phrases ; 7 résumés de référence)

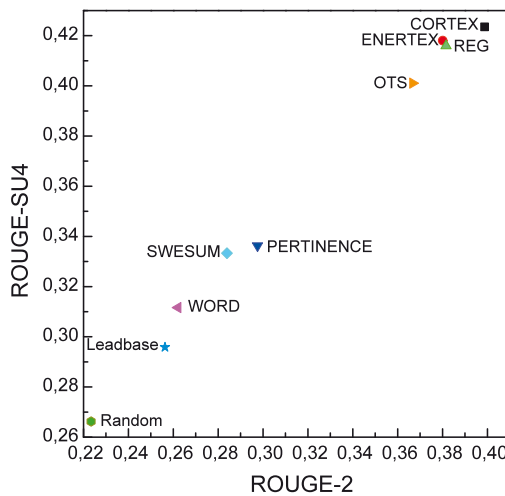


Figure 5 : Rouge-2 et SU4 pour le corpus en espagnol « Medicina Clinica ». 8 textes (409 phrases, 10 961 mots ; résumé à 11 phrases ; 8 abstracts de référence)

5. Conclusion

Nous avons introduit un algorithme glouton basé sur des approches de graphes. Cela nous a permis de développer un nouvel algorithme de résumé automatique. Des tests effectués ont montré que notre algorithme est efficace pour la recherche de segments pertinents. On obtient des résumés équilibrés où la plupart des thèmes sont abordés dans le condensé final. Les avantages supplémentaires des résumés sont plusieurs : ils sont indépendants de la taille du texte, des sujets abordés, d'une certaine quantité de bruit et de la langue (sauf pour la partie pré-traitement). Les résultats ici présentés sont très encourageants. Nous réservons aussi une expérience d'évaluation sur des résumés tronqués à un nombre fixe de mots. Ceci lisserait le biais de segmentation par phrase induit par les systèmes TALNE selon des critères arbitraires. Nous pensons que l'algorithme glouton REG pourrait être incorporé au système Cortex, où il jouerait le rôle d'une métrique pilotée par un algorithme de décision. Ceci permettrait d'obtenir des résumés à l'aide d'une requête de l'utilisateur ou des résumés multi-documents. Une autre voie intéressante consiste à introduire un vecteur des termes d'un texte décrivant une thématique (topique) qui sera introduit dans le graphe du document. Ainsi, les phrases du document pourraient, ou non, s'aligner selon leur degré de pertinence par rapport à la thématique. Ceci permettrait de générer des résumés personnalisés, telles que définis dans les tâches TAC/DUC. L'approche de graphes orientés sera aussi considérée à l'avenir pour créer une espèce de chaîne « conceptuelles » entre les phrases.

Annexe : Texte Mars

Titre : Un robot sur Mars prend les plus belles photos à ce jour

- [0] Ce début d'année 2004 est l'occasion d'une initiative sans précédent en matière d'exploration spatiale.
- [1] Un effort conjoint des États-Unis et de l'Europe a conduit à l'atterrissage de deux engins sur la surface de Mars, à quelques jours d'intervalle.
- [2] Ces deux robots ont néanmoins une mission un peu différente.
- [3] Le « lander » européen, Beagle 2, qui a touché le sol de Mars le jour de Noël, est destiné à étudier la géologie et l'atmosphère de la planète rouge à partir d'une position statique sur son site d'atterrissage.
- [4] Beagle 2 a été construit au Royaume-Uni pour le compte de l'Agence Spatiale Européenne (ESA).
- [5] Malheureusement, aucune communication n'a encore pu être établie avec l'engin à l'heure de la rédaction de ces lignes.
- [6] Les efforts se poursuivent.
- [7] Le second lander, Mars Exploration Rover Spirit (« MER », ou « Spirit »), envoyé par la NASA, a atterri le 3 janvier dernier.
- [8] Contrairement à son malheureux homologue européen, il est conçu pour se déplacer sur le sol de Mars, un peu à la manière du célèbre Sojourner, de la mission Pathfinder, mais dans un rayon d'action bien plus important que ce dernier.
- [9] C'est Spirit qui a envoyé ses premières photos du paysage martien hier, avec un niveau de détails et une résolution jamais atteints à ce jour.
- [10] Même les fameux engins Viking 1 et 2, pourtant producteurs d'images d'une extraordinaire beauté en 1976, risquent fort de sombrer dans l'oubli.

Remerciements

Ce travail a été réalisé, en partie, pendant le séjour de Javier Ramirez comme professeur invité au Laboratoire Informatique d'Avignon (financement UAPV-BQR) en juillet 2009. Nous remercions également le projet RPM2 (ANR France), qui a financé partiellement le projet.

Références

- Boudin F. and Torres-Moreno J.M. (2007). NEO-CORTEX : a performant user-oriented multi document summarization system. In *CICLing '07*, Mexico DF., Springer LNCS Proceedings 4394, pp. 551-562.
- Boudin F. et Torres-Moreno J.M. (2009). Résumé automatique multidocument et indépendance de la langue : une première évaluation en français. In *TALN'09* Senlis, France.
- Boudin F., Torres-Moreno J.M. and Velazquez-Morales P. (2008). An Efficient Statistical Approach for Automatic Organic Chemistry Summarization. In *6th International Conference on Natural Language Processing, GoTAL*, 89-99, Springer, Lecture Notes in Computer Science, 5221, Gothenburg, Sweden.
- Brin S. and Page L. (1998). The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems*, Elsevier Science Pub. B. V., pp. 107-117.
- Edmundson H.P. (1969). New Methods in Automatic Extracting. *Journal of the ACM (JACM)*, 16, 2 : 264-285.
- Erkan G. and Radev D.R. (2004). LexRank : Graph-based Lexical Centrality as Saliency in Text Summarization. *Journal of Artificial Intelligence Research*, vol. (22) : 457-479.
- Fernandez S., SanJuan E. and Torres-Moreno J.M. (2008). Enertex : un système basé sur l'énergie textuelle. *TALN'08*, Avignon, France, pp. 99-108.
- Gould R. (1988). *Graph Theory*. Menlo Park (CA) : The Benjamin/Cummings Publishing Company, Inc.
- Inderjeet M. and Mark T.M. (1999). *Advances in Automatic Text Summarization*. Cambridge (MA): The MIT Press.
- Kintsch W. and van Dijk T.A. (1978). Toward a model of text comprehension and production. *Psychological Review*, vol. 5, 85 : 363-394.
- Lin C.-Y. (2004). Rouge : A package for automatic evaluation of summaries. In Moens, M.-F. and Szpakowicz, S., editors, *Text Summarization Branches Out : Proceedings of the ACL-04 Workshop*, pp. 74-81.
- Luhn H.P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, vol. 2, 2 : 159-165.
- Mani I. and Maybury M.T. (1999). *Advances in Automatic Text Summarization*. Cambridge: The MIT Press.
- Mihalcea R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *ACL'04 on Interactive poster and demonstration sessions*, Association for Computational Linguistics, Morristown, NJ, USA., pp. 181-184.
- Mihalcea R. (2005). Language Independent Extractive Summarization. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, Ann Arbor, Michigan, Association for Computational Linguistics, pp. 49-52.
- Pollock J.J. and Zamora A. (1975). Automatic Abstracting Research at Chemical Abstracts Service. *Journal of Chemical Information and Computer Sciences*, vol. 15, 4 : 226-232.
- Porter M.F. (1980). An Algorithm for suffix stripping. *Program*, vol. 3, 14 : 130-137.
- Radev D., Winkel A. and Topper M. (2002). Multi Document Centroid-based Text Summarization. In *ACL 2002*, Philadelphia, PA, July.
- Torres-Moreno J.M., Velazquez-Morales P. and Meunier J.G. (2002). Condensés de textes par des méthodes numériques. In Morin, A. et Sébillot, P., Morin A. et Sébillot P. editors, *JADT'02*. IRISA/INRIA, France, vol. 2 : 723-734.
- Van Dijk, T.A. (1979). Recalling and summarizing complex discourse. In Burchart, W. and Hulker, K., editors, *Text Processing*, Berlin : Walter de Gruyter, pp. 49-93.