Un automate pour évaluer la nature des textes

Hubert Marteau*, Nicole Vincent**

*Laboratoire d'Informatique, 64 av Jean Portalis, 37200 Tours
hubert.marteau@etu.univ-tours.fr
http://www.li.univ-tours.fr
**Laboratoire CRIP5-SIP, Université Paris 5, 45 rue des Saints Pères, 75270 Paris Cedex 06
nicole.vincent@math-info.univ-paris5.fr
http://www.math-info.univ-paris5.fr/crip5/

Résumé. On ne peut s'intéresser aux textes sans s'intéresser à leur nature. La nature des textes permet de distinguer les textes d'un point de vue primaire. Elle est utilisée pour identifier les textes artificiels, pour la reconnaissance de la langue, afin d'identifier les SPAMS … En ce sens, la méthode la plus connue reste encore la méthode de Zipf. Cet article propose une nouvelle méthode basée sur un automate. L'automate construit un signal pour chaque texte. L'automate est présenté en détail et des expérimentations montrent son utilité dans les domaines aussi divers que ceux cités précédemment.

1 Introduction

L'indexation de textes consiste à trouver une représentation vectorielle d'un texte. Ce vecteur contient les caractéristiques propres au texte et il est, la plupart du temps, utilisé pour permettre une recherche rapide de textes ou d'informations présentes dans les textes.

La représentation la plus commune des textes est le vecteur de fréquence Salton (1989). Le passage du corpus à une telle représentation crée donc une matrice à deux dimensions. Les colonnes représentent les documents. Les lignes représentent les différents mots du corpus. Chaque valeur de la matrice indique le nombre de fois où le mot apparaît dans le texte.

Cavnar et Trenkle (1994) utilisent les vecteurs de fréquence avec les n-grammes de caractères et non les mots. Labbé et Labbé (2001) étudient des fréquences normalisées. Ils espèrent ainsi ôter le biais amené par les différences de longueur des textes. Mothe et al. (2001) proposent, pour améliorer l'indexation, de définir une représentation en trois dimensions. La troisième dimension sert à représenter des informations de type structurel (balises). De Vel (2000) représente les textes essentiellement par leurs caractéristiques structurelles : nombre de total de mots, longueur moyenne des mots, nombre de phrases, …, fréquence d'apparition de mots outils.

L'indexation fait parfois intervenir des méthodes liées davantage au traitement de la langue naturelle Rajman et Besançon (2004). Ainsi, Da Sylva (2004) ne se contente pas uniquement des mots, mais ajoute à la représentation des termes (préfixes, suffixes, …), des paires de mots, …

Cependant d'autres choisissent une autre modélisation. SanJuan et Ibekwe-SanJuan (2002) représentent chaque texte sous forme d'un graphe. Ce graphe est construit à partir des

relations linguistiques que les unités textuelles les plus pertinentes partagent entre elles. Pour cette représentation particulière, les fonctions nécessaires au traitement sont liées à la représentation. Ainsi la classification consiste à trouver les composantes connexes des graphes et à appliquer une Classification Ascendante Hiérarchique avec saut minimum.

Une fois que l'index est réalisé, il peut être utilisé, dans le cadre d'une recherche d'information. Il peut aussi être utilisé pour calculer les distances entre les textes ou comme entrée d'un classifieur. Labbé et Labbé (2001) utilisent la mesure de distance de Bray-Curtis. Salton et Allan (1994) utilisent la mesure du cosinus. Lelu (2002) compare au cosinus, précédemment cité, la mesure du Chi² et le cosinus dans l'espace distributionnel.

Concernant les classifieurs, Morin (2002) utilise une Analyse Factorielle des Correspondances (AFC). Tan (2005) améliore un classifieur des k-plus proches voisins en pondérant les classes suivant le nombre d'individus qu'elles contiennent. Jörgensen (2005) tente d'extraire le contexte d'utilisation des mots (différence entre le lit pour dormir et le lit de rivière) en utilisant un réseau de neurones. Cavnar et Trenkle (1994) calculent, avec une méthode semi-supervisée, un profil pour chaque classe puis la distance de chaque texte à chacun des profils. Shankar et Karypis (2000) calculent les centroïdes des classes et les normalisent. De Vel (2000) opte pour les SVM. Zhang (2000) compare les SVM au perceptron et à des algorithmes Winnow (entre le perceptron et les SVM). Rennie et al (2003) préfèrent les Réseaux Bayésiens. Klopotek (2005) utilise les réseaux Bayésiens sous forme arborée. Sinka et Corne (2004) utilisent les K-means, et Brouard (1999) les Chaînes de Markov Cachées.

La classification de textes tient sa spécificité de la représentation qui est faite des données textuelles, c'est-à-dire de l'indexation qui est faite des textes. Le reste du traitement suit un schéma classique de classification. Un schéma identique a déjà été observé, précédemment, pour la classification thématique Forrest et Meunier (2000).

Certaines méthodes sortent de ce schéma. Dans le cadre de la classification thématique, une indexation originale consiste à représenter chaque texte par une image Reynar (1994).La segmentation thématique est réalisée à partir d'une analyse des images.

Ce grand nombre de méthodes amène avant tout une question très importante : qu'est-ce qui est classé ? Ce sont évidemment les textes qui sont classés. Mais il faut y voir une question bien plus profonde : sur quelle base les textes sont-ils classés ?

L'indexation des textes est donc une étape primordiale. C'est la première étape du traitement et elle consiste à passer d'un texte à une forme vectorielle numérique (ou autre). De nombreuses méthodes choisissent de représenter le contenu des textes. De Vel (2000) choisit de représenter leurs informations structurelles.

Ainsi, la classification textuelle se heurte à deux sortes de difficultés. Il y a les difficultés « habituelles » dans le domaine général de la classification. Cela signifie qu'il faut trouver le meilleur système de classification pour les données à traiter. Mais, il y a, avant tout, la difficulté de la représentation des textes du corpus, car c'est sur la base de cette représentation que les textes sont classés.

Cet article se situe dans le cadre du traitement des entretiens sociologiques oraux retranscrits.

Les méthodes actuelles proposent une représentation du contenu à partir de vecteurs de fréquences ou proposent une représentation de la structure à partir de caractéristiques structurelles des textes.

Dans le cas d'entretiens sociologiques oraux retranscrits, la structure est naturellement inexistante, car la forme écrite n'est pas la forme originale.

Le problème du langage naturel oral implique que le contenu ne semble, a priori, pas non plus une solution à l'indexation d'entretiens sociologiques. Ce problème apparaît comme une difficulté dans la représentation des entretiens. En effet, les entretiens utilisent le langage oral courant, sans spécificité de vocabulaire.

Une solution est proposée par cet article. Il est, en effet, émis l'hypothèse que le sens est porté par la structure des textes plus que par leur contenu. Le mot structure est à prendre au niveau le plus proche du texte qui soit. Cela signifie que l'organisation du langage est propre à chacun. Il est donc proposé de classer les entretiens non pas sur leur fond, mais sur leur forme.

Cet article propose de définir un automate pour construire une représentation des textes.

2 Automate 1D

Avant de détailler la méthode qui est basée sur un automate, quelques rappels sont présentés sur les automates finis déterministes. Ces rappels permettent de présenter les bases théoriques des automates Après ces quelques rappels, l'automate et la méthode d'évaluation sont détaillés.

2.1 Rappels sur les automates

D'après Nourredine (1992), de manière intuitive, on peut voir un système de reconnaissance comme une machine permettant de lire un mot à travers différentes manipulations. Cette machine, appelée automate, permet donc, par extension, de reconnaître un langage.

Il existe plusieurs types d'automates, cependant ils ont tous une structure commune. Ainsi, un automate est composé de trois parties :

- Une bande en entrée, finie ou infinie, sur laquelle est inscrit le mot à lire. La bande, en entrée, est divisée en cellules ; le mot à lire étant formé d'une suite de symboles (de l'alphabet), un symbole (et un seul) est logé dans une cellule.
- Un organe de commandes qui permet de gérer un ensemble fini d'états. La gestion des états se fait à travers une fonction spécifique, dite fonction de transition.
- Eventuellement, une mémoire auxiliaire de stockage.

Aho et al (1986), Nourredine (1992) définissent un automate fini non-déterministe comme un modèle mathématique défini par le cinq-uplet A(X,E,e0,t,F), avec :

- X : un ensemble de symboles d'entrée (l'alphabet des symboles d'entrée)
- E : l'ensemble des états
- e0 : un état qui est distingué comme l'état de départ ou état initial
- t : une fonction de transition, qui fait correspondre des couples état-symbole à des ensembles d'états
- F : un ensemble d'états distingués comme états d'acceptation ou états finals.

Un automate fini non déterministe peut être représenté graphiquement comme un graphe orienté étiqueté, appelé graphe de transition, dans lequel les nœuds sont les états et les arcs étiquetés représentent la fonction de transition. Ce graphe ressemble à un diagramme de transition, mais le même caractère peut étiqueter deux transitions ou plus en sortie d'un même nœud et les arcs peuvent être étiquetés par le symbole spécial ε (neutre) au même titre que les symboles d'entrée.

Aho et al (1986), Nourredine (1992) considèrent qu'un automate fini déterministe est un cas particulier des automates finis non déterministes dans lequel :

- aucun état n'a de ε-transition, c'est-à-dire de transition sur l'entrée ε
- pour chaque état e et chaque symbole d'entrée a , il y a au plus un arc étiqueté a qui quitte e.

Un automate fini déterministe a au plus une transition à partir de chaque état sur n'importe quel symbole. Si on utilise une table de transition pour représenter la fonction de transition de l'automate fini déterministe, alors une entrée dans la table de transition est un état unique.

2.2 L'automate

L'indexation par automate 1D est une nouvelle méthode développée dans le cadre de l'étude d'entretiens sociologiques. En effet, dans ce type d'application, le contenu est trop pauvre pour que l'on puisse s'y attacher. Ces textes sont, en effet, des retranscriptions de discours oraux et, dans ce cas, la langue naturelle utilise souvent moins de 1000 mots.

Il a été émis l'hypothèse que le sens peut être porté par la structure des textes. Cela signifie que le contenu du discours influence sa structure. Ainsi, évaluer, d'un point de vue sociologique, les entretiens revient à évaluer leur structure.

Cette méthode d'indexation tente de représenter la structure du texte par l'enchaînement des informations qui le composent. C'est donc la dynamique de la structure qui est mise en évidence. C'est-à-dire que la structure n'est pas représentée de manière globale, mais à partir de ses comportements locaux, à partir de son évolution. Les textes sont une représentation mono-dimensionnelle de l'information. Cet automate tente de conserver cet aspect mono-dimensionnel.

L'appellation « automate 1D » provient donc du fait que l'automate essaie de construire un signal d'amplitude fixée, à partir du texte.

2.2.1 Etats et transitions

Le signal évolue dans un intervalle d'amplitude bornée et discrète. C'est-à-dire que chaque valeur de l'amplitude est représentée par un état. De plus, chaque état n'est relié qu'aux états de valeurs directement supérieure et inférieure. C'est-à-dire que l'automate ne propose que deux types de transitions, la transition « Aller vers le bas » qui permet d'aller dans l'état de valeur directement inférieure et la transition « Aller vers le haut » qui permet d'aller dans l'état de valeur directement supérieure. Les états aux extremums n'ont, naturellement, qu'un seul voisin.

L'ensemble des symboles d'entrée correspond à l'ensemble des unités textuelles pouvant être traitées. Afin de laisser un maximum de liberté à l'automate, il est placé, initialement sur l'état qui se situe au centre de l'intervalle. C'est-à-dire que l'élément e_0 correspond à l'état central. Il a été choisi, afin de ne pas devoir effectuer d'étude préalable des textes du corpus, de ne pas affecter les transitions avant le traitement. C'est-à-dire que les fonctions de transition sont définies au fur et à mesure du traitement. L'affectation suit une liste de règles énoncées dans la partie qui suit.

2.2.2 Les règles d'affectation

L'amplitude maximale du signal est fixée a priori. Il faut éviter au maximum de se rapprocher des limites de l'intervalle et en aucun cas les franchir.

Par contre, la signal n'a d'intérêt que s'il varie. Il faut donc éviter son immobilité.

Les règles ont été constituées afin de répondre à ces contraintes. Elles sont, néanmoins, liées à la distribution naturelle des unités textuelles dans les textes traités.

D'un point de vue localisation, les états sont divisés en quatre classes d'effectifs identiques. Les états périphériques ont été définis comme correspondant au regroupement des deux classes d'états aux limites de l'amplitude, c'est-à-dire la classe la plus en bas et la classe la plus en haut. Les états centraux correspondent aux deux classes centrales.

Voici la liste des différentes règles :

- La première règle concerne le premier symbole d'entrée, c'est-à-dire la première unité textuelle du texte. L'action « Aller vers le haut » lui est assignée. En dirigeant le signal vers les classes périphériques, cela permet d'éviter l'immobilité initiale du signal.
- Lorsque l'automate est dans un état périphérique, la seconde règle impose de suivre la direction du centre de l'amplitude aux symboles d'entrée auxquels aucune transition n'a été affectée. Ainsi, si il arrive un symbole d'entrée auquel aucune action n'a été affectée et que l'automate se situe dans le quart bas des états, alors, il est assigné au symbole la transition « Aller vers le haut ». Cette règle essaie, ainsi, de donner au signal créé une allure sinusoïdale.
- La troisième règle indique le cas général : si aucune transition n'est affectée à un symbole d'entrée et que ce symbole n'est concerné par aucune des premières règles, alors il lui est affecté la même transition que celle qui est affectée au symbole qui le précède. Cette troisième règle assure, ainsi, un minimum de continuité et de cohésion dans les déplacements effectués et donc dans la formation du signal.
- Enfin la quatrième et dernière règle, dite règle de changement d'urgence, inverse l'action d'un symbole d'entrée lorsque l'action qui lui est assignée implique la sortie de l'intervalle.

La **FIG. 1** montre l'action de l'automate sur deux textes différents comme s'il était enregistré par un outil du type sismographe. Le premier texte est un entretien sociologique. Le second texte est un texte artificiel conçu statistiquement à partir des fréquences d'apparition des caractères en langue française. Dans la figure, le centre et les quarts sont marqués par des barres horizontales. L'amplitude a été, ici, fixée à 301.

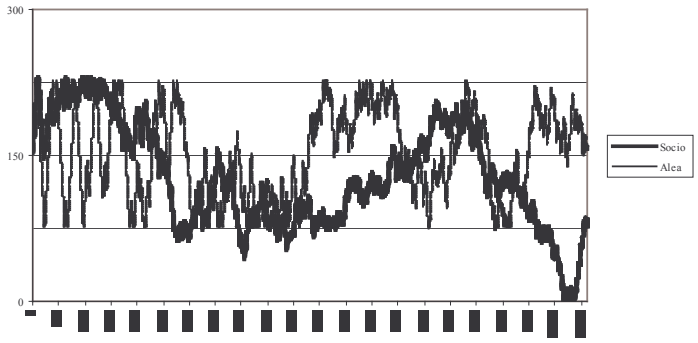

FIG. 1 - *Action de l'automate sur un texte sociologique et un texte généré aléatoirement*

Le texte artificiel est nettement marqué par son aspect statistique et reste enfermé dans la partie centrale. Le texte en langue naturelle se différencie par une oscillation générale beaucoup plus lente. Il se caractérise surtout par des unités textuelles formatées pour descendre et cela jusqu'aux limites de l'automate, qu'il atteindra 13 fois au total.

L'automate représente donc réellement l'évolution du texte.

2.3 Valorisation

Lorsque l'on traite un signal comme celui fourni par l'automate, on peut se poser la question de la valorisation. Partant de l'hypothèse que l'amplitude générale d'un signal dépend de la structure du texte auquel il correspond, il est proposé d'étudier la façon dont les états sont utilisés. Pour cela, l'écart-type du signal est calculé.

Chaque texte est donc représenté numériquement par l'écart-type du signal qui le représente.

3 Expérimentations

Cette partie présente quatre types d'expérimentations. La première série s'intéresse aux codages afin de montrer que l'automate reflète réellement la structure des textes. La deuxième série étudie différentes œuvres de la littérature française. Cela montre, que dans le cadre d'une étude littéraire, l'automate permet de distinguer les différents styles de textes. La troisième série d'expérimentations s'attache à montrer la différence entre des textes courts écrits en anglais et des textes courts écrits en français. Enfin, la quatrième série montre l'utilité de l'automate pour différencier les messages électroniques.

La **FIG. 3** présente l'évolution des écarts-types suivant la taille du vecteur sur différents codages. Le corpus est constitué de 4 types de textes : 13 textes générés de manière complètement aléatoire, 16 textes représentant des sons de déglutition, 19 textes de fichiers midi et

27 fichiers de programmation. Pour un meilleur affichage, une échelle logarithmique est utilisée.

Il peut, tout d'abord, être observé que l'évolution de l'écart-type est similaire pour tous les codages. Il n'y a qu'en utilisant une amplitude de 3 que les positions relatives sont changées.

La figure représente le caractère répétitif des fichiers textes générés aléatoirement. Les valeurs sont comprises entre 2100 pour une amplitude de 5 et 18 pour une amplitude de 1001 cases.

Les sons de déglutition montrent, avec des valeurs allant, pour les mêmes amplitudes, de 3100 à 28, une structure plus complexe. Cependant, les fichiers textes de sons de déglutition représentent les informations d'évolution locale de chaque signal sonore. C'est-à-dire que les codes indiquent si, localement, le signal monte ou si il descend. Il paraît donc logique que ce fichier soit répétitif.

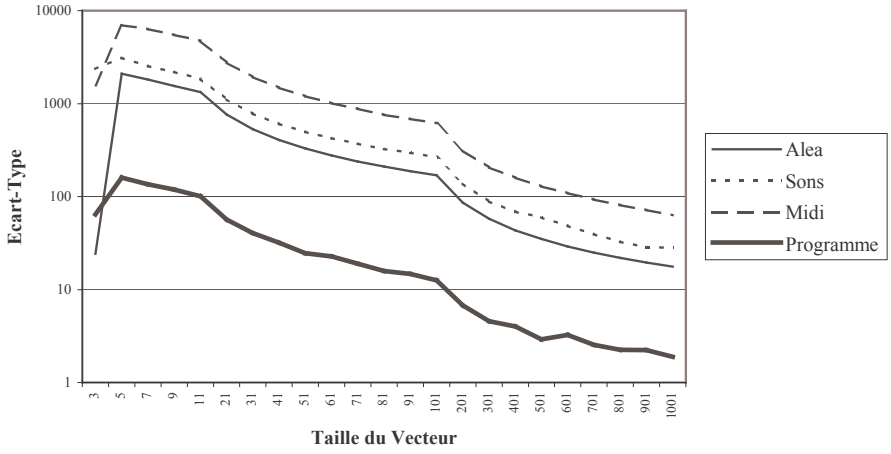

FIG. 3 - *Evolution de l'écart-type suivant la taille du vecteur pour des codes*

Il en va de même pour les fichiers midi. Les fichiers midi représentent des partitions de musique. C'est-à-dire qu'ils contiennent la portée de chaque instrument. La nature même de la musique veut que les phrases musicales soient répétées régulièrement. Cela se ressent dans le fichier texte et donc dans la structure du signal construit par l'automate. Néanmoins, les phrases musicales sont suffisamment longues pour que la répétition des informations ne paraisse pas artificielle. Ainsi, pour une amplitude de 5, l'écart-type est de 7000 et, pour une amplitude de 1001, l'écart-type est de 63. Ces valeurs sont le triple des valeurs des fichiers aléatoires et le double des valeurs des sons de déglutition.

Inversement, un langage de programmation ne propose qu'un vocabulaire très limité et répété à l'extrême. On y trouve, en effet, en permanence des déclarations de fonctions, des déclarations des types de variables, des noms de variables utiles « i », « j », des boucles

« for », « while », … Cette répétition se représente par des valeurs d'écart-type très faibles : 159 pour une amplitude de 5 et 2 pour une amplitude de 1001.

La **FIG. 5** s'intéresse à l'évolution de l'écart-type pour des textes littéraires français. Le corpus est constitué des œuvres de cinq auteurs différents. Il y a 34 œuvres de Corneille, 32 œuvres de Molière, 8 œuvres de Maupassant, 12 œuvres de Racine et 7 œuvres de Zola.

Comme pour les codages, on distingue différents types de structures. Ce sont les pièces de théâtre de Corneille qui ont le caractère le plus répétitif. Avec un écart-type d'à peine 2500 pour amplitude de 5 et 26 pour une amplitude de 1001, ces œuvres suivent la structure des sons de déglutition.

Les œuvres de Racine et Molière sont mêlées. Cela signifie que l'écriture est plus libre et que le style est plus varié. Les valeurs d'écart-type sont 7000 pour une amplitude de 5 et 68 pour une amplitude 1001, c'est-à-dire la même évolution que les fichiers midi vus précédemment.

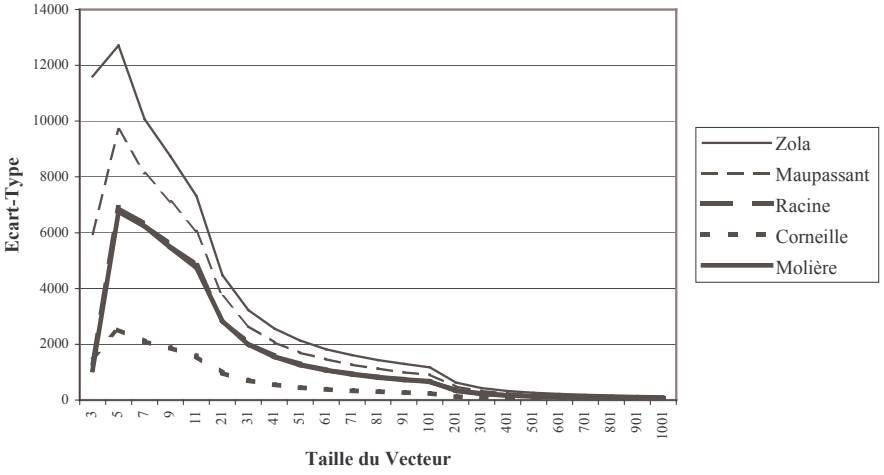

FIG. 5 - *Evolution de l'écart-type suivant la taille du vecteur pour des textes littéraires*

Avec les œuvres de Guy de Maupassant, on passe du théâtre aux romans. Le style est à nouveau plus libre. L'écart-type s'en ressent. Il est de 10000 pour une amplitude de 5 et de 100 pour une amplitude de 1001. Une analyse plus détaillée des résultats sur les œuvres de Guy de Maupassant met en évidence une distinction entre les œuvres *L'Âme Etrangère* et *L'Angélus*, d'un côté et les autres œuvres de l'autre côté. Cette distinction avait déjà été observée par d'autres études Marteau et al (2003) et Marteau et al (2005).

Enfin, les œuvres de Zola, réputées pour être une description du 19^{e} siècle, suivent une structure complètement libre. L'écart-type moyen va de 13000 pour une amplitude de 5 et 135 pour une amplitude de 1001. L'œuvre *J'accuse* se différencie en suivant une évolution

plus proche de celle de Molière et Racine. De même, l'œuvre *Germinal* se distingue des autres œuvres avec un style beaucoup plus libre et une structure beaucoup moins marquée (16500 pour une amplitude de 5 et 180 pour une amplitude de 1001).

La **FIG. 7** présente les résultats obtenus sur deux corpus de textes courts : Amaryllis et NewsGroups. Le premier corpus est constitué de textes de type journalistique écrits en français, le second corpus est constitué de textes issus de NewsGroups écrits en anglais. A priori, on peut supposer de grandes différences dans les résultats.

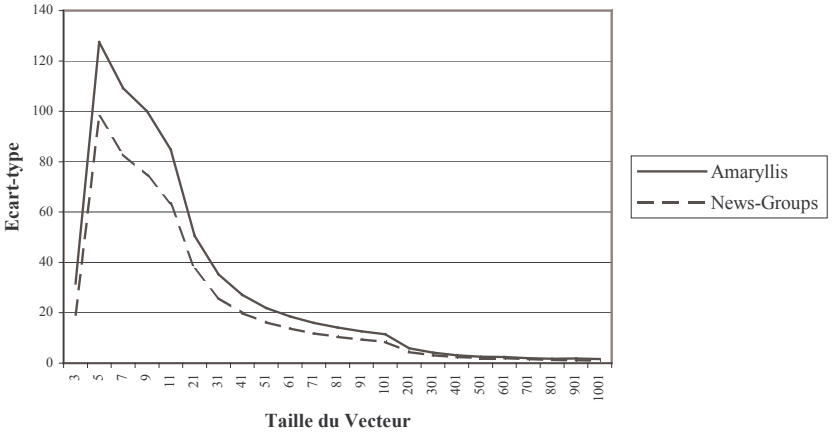

FIG. 7 - *Evolution de l'écart-type suivant la taille du vecteur pour textes courts de langues différentes*

En effet, les deux styles semblent limités dans l'utilisation qui est faite du vocabulaire : les articles concernent un évènement en particulier, les messages de NewsGroups concernent un thème en particulier. Cependant, la structure de langage à la base de chacun des corpus, est différente. En anglais, les verbes suivent en majorité une même forme et la langue est réputée pour admettre les répétitions. En français, c'est le contraire, les verbes ont, la plupart du temps, des formes différentes et les répétitions ne sont que très peu appréciées.

Le fait de travailler sur des textes courts réduit les valeurs d'écart-type. Les valeurs sont, en effet, similaires à celles obtenues par le corpus programme de la **FIG. 3**, les textes des corpus Amaryllis (en moyenne 330 mots) et NewsGroups (en moyenne 355 mots) sont pourtant, en moyenne 2 fois plus courts que ceux du corpus Programme (en moyenne 660 mots).

Les évolutions montrées par la **FIG. 7** confirment les a priori formulés précédemment. La structure du Corpus Amaryllis semble, en effet, plus libre que celle du corpus NewsGroup. Les écarts-types sont, respectivement, 130 et 100 pour une amplitude de 5 et 1,6 et 1 pour une amplitude de 1001.

Enfin, la FIG. 9 présente des résultats obtenus sur deux types de messages électroniques. Le premier corpus est un ensemble de 51 mails divers concernant l'inscription à des conférences, les notifications d'acceptation, des informations pédagogiques d'une école d'ingénieurs (Polytech'Tours Département Informatique), des informations internes à l'équipe de recherche et des communications personnelles (humour, relations humaines, …). Le second corpus est constitué de messages qui polluent les boîtes mails chaque jour : les Spams. Les textes sont des textes publicitaires concernant des logiciels pirates, du Viagra, du XXX et autres messages sans intérêt.

Les messages électroniques sont des textes encore plus courts que ceux des corpus Amaryllis et NewsGroups. Certains messages, mails ou spams, sont de l'ordre de deux ou trois phrases. L'automate arrive néanmoins à distinguer, d'une manière générale, les deux types de messages. En détail, il apparaît que dans les textes trop courts, par exemple ceux qui contiennent une phrase et une adresse mail, l'automate n'arrive pas à distinguer lesquels sont mails et lesquels sont spams. Dans le corpus constitué, cela concerne 20% des messages.

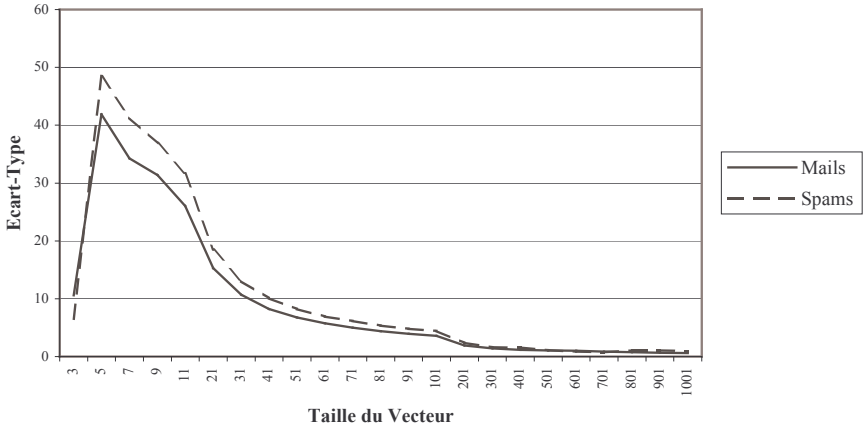

FIG. 9 - Evolution de l'écart-type suivant la taille du vecteur pour des messages électroniques

4 Conclusion

Cet article a donc proposé une nouvelle méthode qui tente de représenter un texte par sa structure, ou plutôt par son évolution. C'est en effet la séquence particulière des mots, leur ordre, … qui permet de représenter le texte.

Quoiqu'il en soit, cette méthode est, comme la méthode de Zipf, détachée du contenu, c'est-à-dire que les textes ne sont pas comparés par rapport à leur contenu. Cette méthode est composée d'un nombre fixé auparavant de caractéristiques et ce nombre est bien inférieur à ceux de la méthode vectorielle et de la méthode de Zipf. Et cette méthode assure de meilleur résultat que la méthode de Zipf.

Cela indique donc que l'évolution, si elle est correctement capturée pourrait servir à représenter les textes.

Références

Aho, A., R. Sethi and J. Ullman (1986). *Compilers*, Addison Wesley.

Brouard, T. (1999). *Algorithmes Hybrides d'Apprentissage de Chaînes de Markov Cachées : Conception et Application à la Reconnaissance des Formes*, Rapport de Thèse de Doctorat, Université François-Rabelais de Tours, 219 pages.

Cavnar, W. B. and J. M. Trenkle (1994). *N-Gram-Based Text Categorization*, Third Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, 161-175.

Da Sylva, L. (2004). *Indexation Automatique de Documents par Contribution d'Analyses Statistiques et Terminologiques Structurées*, RIAO 2004, Avignon.

De Vel, O. (2000). *Mining E-mail Authorship*, Workshop on Text Mining, KDD-2000, Boston.

Forest, D. et J.G. Meunier (2000). *La Classification Thématique des Textes : un Outil d'Assistance à la Lecture et à l'Analyse de Textes Philosophiques*, 5èmes Journées internationales d'Analyse statistique des Données Textuelles, Lausanne.

Jörgensen, P (2005). *Incorporating Context in Text Analysis by Interactive Activation with Competition Artificial Neural Networks*, Information Processing and Management, 41/5: 1081-1099.

Klopotek, M. (2005). *Very Large Bayesian Multinets for Text Classification*, Future Generation Computer Systems, 21/7:1068-1082.

Labbé, C. and D. Labbé (2001). *Inter-Textual Distance and Autorship Attribution Corneille and Molière*, Journal of Quantitative Linguistic, 213-231.

Lelu, A. (2002). *Comparaison de Trois Mesures de Similarités utilisées en Documentation Automatique et Analyse Textuelle*, 6èmes Journées internationales d'Analyse statistique des Données Textuelles, St Malo.

Marteau, H., A. Lefèvre et N. Vincent (2003). *Comparaison de Textes par Mesure Fractale*, Majestic'03, Marseille.

Marteau, H. et N. Vincent (2005), *L'automate Textuel pour la prise en compte de l'Evolution du Texte*, EGC 05, Paris.

Morin, A. (2002). *Deux Exemples d'Analyse de Données Textuelles*, Colloque sur la statistique et l'analyse des données dans les sciences appliquées et économiques, Beyrouth.

Mothe, J., C. Chrisment, T. Dkaki, B. Dousset and D. Egret (2001), *Information Mining : Use of the Document Dimensions to Analyse Interactively a Document Set*, European Colloquium on Information Retrieval Research, Darmstadt.

Nourredine, M. (1992). *Théorie des Langages*, Office des Publications Universitaires.

Rajman M. and R. Besançon (2004). *Text Mining – Knowledge Extraction from Unstructured Textual Data*, International Federation of Classification Societies, Chicago.

Rennie, J.D.M., L. Shih, J. Teevan and D.R. Karger (2003), *Tackling the Poor Assumptions of Naive Bayes Text Classifiers*, ICML 2003, Washington DC.

Reynar, J.C. (1994). *An Automatic Method of Finding Topic Boundaries*, 32nd Annual Meeting of the Association for Computational Linguistics, Student Session, 331-333, Las Cruces.

Salton, G. (1989). *Automatic Text Processing – The Transformation Analysis and Retrieval of Information by Computer*, Addison Wesley Publishing Company, Reading, MA,.

Salton, G. and J. Allan (1994), *Automatic Text Decomposition and Structuring*, RIAO'94, Paris.

SanJuan, E. et F. Ibekwe-SanJuan (2002). *Terminologie et Classification Automatique des Textes*, 6èmes Journées internationales d'Analyse statistique des Données Textuelles, St Malo

Shankar S. and G. Karypis (2000), *A Feature Weight Adjustment Algorithm for Document Categorization*, Workshop on Text Mining, KDD 2000, Boston.

Sinka M.P. and D.W. Corne (2004). *The BankSearch Web Document DataSet : Investigating Unsupervised Clustering and Category Similarity*, Journal of Network and Computer Applications, 28: 129-146.

Tan, S. (2005), *Neighbor-Weighted K-Nearest Neighbor for Unbalanced Text Corpus*, Expert System With Application, Elsevier, 28/4: 667-671.

Zhang, T. (2000), *Large Margin Winnow Methods for Text Categorization, Workshop on Text Mining*, KDD 2000, Boston.

Summary

We cannot be interested in texts without being interested in their nature. The nature of texts allows to distinguish the texts from a primary point of view. It is used to identify the artificial texts, for the recognition of the language, to identify the SPAMS … This way, the most known method still remains the method of Zipf. This article proposes a new method based on a machine. The machine builds a signal for every texts. The machine is presented in detail and experiments show its utility in domains so different as those quoted previously.