MATHÉMATIQUES ET SCIENCES HUMAINES

NICOLAS TURENNE

Apprentissage d'un ensemble pré-structuré de concepts d'un domaine : l'outil GALEX

Mathématiques et sciences humaines, tome 148 (1999), p. 41-71

<http://www.numdam.org/item?id=MSH_1999__148__41_0>

Math. Inf. Sci. hum., (37e année, n° 148, *1999*, pp. 41-71)

APPRENTISSAGE D'UN ENSEMBLE PRÉ-STRUCTURÉ DE CONCEPTS D'UN DOMAINE : L'OUTIL GALEX

Nicolas TURENNE[1]

RÉSUMÉ – *La quantité d'information textuelle augmente de façon exponentielle aussi bien comme archives que documents de travail dans les organisations académiques, dans les administrations et dans les entreprises. Une solution pour structurer cette montagne de données textuelles est de construire un modèle de connaissances pour indexer cette information. L'acquisition de connaissances doit permettre d'extraire et classifier les données pour aboutir à une indexation conceptuelle. Traditionnellement, les méthodes de classification d'analyse de données étaient adaptées pour des tables classiques de données de la forme objet/attribut/valeur. Nous présentons Galex (Graph Analyzer for LEXicometry) qui développe une structuration de la connaissance grâce à une méthode de clustering de termes. Cette structuration a pour but de synthétiser le contenu d'information présentant un intérêt majeur dans des applications de filtrage d'information ou de navigation hypertextuelle sur des documents similaires. Galex prend en compte la nature des données sur lesquelles il s'applique : le langage naturel. La complexité du langage naturel est bien connue : ambiguïté de sens, constructions grammaticales multiples de la phrase, style, création de termes... Nous montrons qu'à travers l'intégration de notions mal définies mais utiles telles que «concept», «ontologie» et «corpus», le clustering peut être amélioré par adjonctions de connaissances linguistiques. Nous basons notre approche sur des phénomènes typiques tels que des relations graphe-statistiques entre termes, des relations de schéma dans un contexte et la réduction canonique de formes variantes.*

MOTS-CLÉS – Clustering de termes, acquisition de connaissances, ontologie, apprentissage de concepts, analyse de corpus, text-mining, fouille de texte, analyse de données.

SUMMARY – Learning of the pre-structured concept set of a domain: the Galex tool

The huge amount of electronic textual information increases exponentially just as easily as archives and working documents in academic organizations, in administration and in firms. A solution for structuring this mountain of textual database is to build a knowledge model to index this information. One way can be obtained by data extraction and classification producing conceptual indexing by knowledge acquisition. Traditionally the classification methods of Data Analysis were adapted while used for the classical table of data under an object/characteristics/value format. We present Galex (Graph Analyzer for LEXicometry) which develops structuration of knowledge by a term clustering method. This structuration synthetizes the content of information providing the mapping data to information filtering or hypertextual navigation on similar documents. Galex aims at taking into account the nature of the data to which it is applied : natural language. The complexity of natural language is well known: sense ambiguity, multiple grammatical construction of sentence, style, term creation...We show through integration of poorly

[1] Laboratoire d'Informatique et d'Intelligence Artificielle (L.I.I.A.), ENSAIS, Université Louis-Pasteur, 24, boulevard de la Victoire, 67000 Strasbourg, e-mail : turenne@liia.u-strasbg.fr, http://www-ensais.u-strasbg.fr/LIIA/liia.htm

defined, though useful as concept, ontology, term and corpus, notions that clustering can be improved by adding linguistic knowledge. We base our approach on typical phenomena such as graph-statistical relations between terms, scheme relations in a context and canonical reduction of variants.

KEYWORDS – Terms Clustering, Knowledge Acquisition, Ontology, Concept Learning, Corpus Analysis, Text-Mining, Statistical Data Analysis.

1. INTRODUCTION

Depuis la naissance des méthodes de classification automatique, le schéma standard de données variable/attribut/valeur a été beaucoup exploité. Les domaines étudiés étaient la zoologie et la botanique où les attributs pouvaient être extraits comme colonnes de données (voir Tableau 1). Dans cet exemple, nous pouvons extraire deux classes d'oiseaux : (autruche, flamant rose) et (poule, pigeon) en fonction de leurs caractéristiques sémantiques.

	petit	grand	forêt	Village	steppe
Pigeon	1	0	1	1	0
Autruche	0	1	0	0	1
Poule	1	0	0	1	0
Flamant rose	0	1	0	0	1

Tableau 1. Représentation classique d'objets

La représentation multivariée est particulièrement bénéfique en Analyse de Données. Plusieurs méthodes sont capables de réaliser des traitements efficaces : la classification hiérarchique agglomérative, l'analyse factorielle des correspondances, l'analyse relationnelle (Lebart, Salem & Berry, 1998) et même des méthodes non-linéaires comme les réseaux de neurones (Kohonen, 1989 ; Lingras, 1994 ; Memmi, Gabi & Meunier, 1998). Elles donnent approximativement les mêmes résultats (Meila & Heckerman, 1998 ; Schütze & Silverstein, 1997 ; Messatfa & Zait, 1997 ; Turenne & Rousselot, 1998).

Mais dans le traitement du langage naturel, la structure des données est implicite et n'est pas donnée par un ensemble d'attributs. À l'heure actuelle, aucune approche linguistique ne peut apporter de théorie unifiée de la sémantique. Notamment, une telle théorie pourrait, en informatique, prétendre décrire tous les mots, expressions et phrases sans ambiguïté de sens. Des tentatives réussies ont été réalisées pour des domaines fermés spécifiques (avec des centaines de mots simples), la plupart du temps concernant des sous-langages techniques : aéronautique, droit, ..., mais jamais pour la totalité de la connaissance exprimée en langage naturel. Ce traitement partiel freine l'adaptation des méthodes usuelles de classification à n'importe quelle partie de données textuelles ou de réseau de données textuelles en entreprise (internet, groupware, étude statistique qualitative, notices bibliographiques...). D'un autre côté, les méthodes statistiques sont basées sur des métriques assurant la comparaison entre éléments (Cutting & Karlgren, 1996). Ces métriques sont dans tous les cas établies par rapport à une unité d'un espace

algébrique et, dans le cas d'une application en langage naturel, cette unité n'a pas d'équivalent en termes de sémantique du langage. Aujourd'hui, aucune description d'une unité de distance sémantique n'a été développée. En fait, une unité métrique dépend du point de vue de l'espace sémantique. Par exemple dans une base de données commerciale, un client et l'historique d'un client sont proches l'un de l'autre mais dans l'organisation fonctionnelle de la société un client est associé aux catégories externes (comme la commande) et l'historique d'un client est associé aux catégories de pilotage (comme les ressources humaines). Les méthodes de classification utilisant une métrique montrent que la comparaison de la similarité par rapport à une métrique aboutit à d'intéressantes «classes»[2]. Nous pouvons déduire une corrélation d'un terme dans le contexte distributionnel d'autres termes. Nous exploitons cette observation avec une approche développant un clustering théorique par graphe. Le clustering est connu pour grouper des objets en ensembles homogènes. L'interprétation des groupes est très proche de ce que l'on considère comme une catégorie ou un concept, c'est pourquoi la classification est largement utilisée dans la catégorisation de documents (Bisson, 1996). Maintenant nous allons discuter de plusieurs notions nécessaires à la construction d'un réseau final : un concept/catégorie (i.e. classe), un item cible d'une catégorie (instances et relations), le media d'une catégorie (documents). Dans la deuxième partie de l'article, nous présentons toutes ces notions encore insuffisamment établies pour être formalisées et restant la cible de discussions animées dans les communautés de recherche. Dans la troisième partie, nous présentons l'extraction de termes ou unités sémantiques supposées que nous visons à regrouper en catégories. Dans la quatrième partie, nous présentons la méthode de clustering. Enfin, dans la cinquième partie, nous présentons les tests et discussions avec certains échantillons de données.

2. CADRE DE L'ÉTUDE

Notre but est de synthétiser la connaissance de données textuelles dans un réseau liant des groupes de termes aussi proches que possible de catégories. Nous allons décrire plusieurs notions utilisées pour atteindre notre objectif à travers le clustering. Ces notions sont : la terminologie, le concept et le corpus de texte. Premièrement, nous devons noter quelques lacunes dans la formalisation de ces notions dues à l'état de l'art des théories linguistiques et modèles d'ingénierie des connaissances. Les discussions ne permettent pas d'imposer un consensus sur la définition universelle de ces notions.

2.1. TERME ET SENS

Depuis le début du siècle (Wüster, 1991 ; Frege, 1892), la terminologie a justifié son rôle pour définir la connaissance d'un domaine grâce à des expressions linguistiques spécifiques et des dénotations du monde réel associées à un formalisme logique servant à décrire les concepts. Un terme n'est pas spécialement un vecteur de sens par des connaissances linguistiques intrinsèques. Nous pouvons distinguer deux sortes de vecteurs sémantiques. Premièrement, le mot pointe vers un objet du monde réel. Le sens est une bijection entre deux espaces : l'espace linguistique des mots et l'espace du monde réel. Nous appelons ce phénomène la référence d'un mot (exemple : «carte bleue»). Deuxièmement le mot prend son sens à partir du contexte entourant le mot. Nous appelons ce phénomène le sens d'un mot. Ces deux formes de sens révèlent la complexité

[2] Que l'on nommera, dans la suite de l'article, par l'américanisme «clusters» – en français «cliques» ; de même, on dira «clustering» pour classification (N.d.l.R.).

de l'expressivité du langage naturel divisée en construction dynamique et construction statique de sens. La construction du sens n'est pas seulement lexicale ou syntaxique, car la sémantique agglomère des facteurs non-linguistiques tels les phénomènes visuels, tactiles et également tous les phénomènes pragmatiques. L'origine de ces phénomènes pragmatiques ne provient pas de l'espace linguistique. Nous les appelons phénomènes «extrasémantiques».

Ces phénomènes «extrasémantiques» influent sur les facteurs de compréhension. Ainsi le contexte est une forme fondamentale de la construction du sens mais ne se réduit pas aux sources linguistiques et textuelles. D'un autre côté, le texte est le media d'archivage le plus accessible pour l'acquisition des connaissances depuis que la mémoire de notre connaissance est transformée en documents électroniques (Aussenac-Gilles, Bourigault & Condamines, 1995 ; Skuce & Meyer, 1991). Nous réunissons des unités de sens dont ce que nous espérons être des termes du domaine. Pour nous, un terme est un syntagme nominal représentatif d'un domaine. Un terme doit contenir une interprétation sémantique non ambiguë et contrainte par le domaine qu'il caractérise. Cependant un terme peut être décliné en un certain nombre de formes de significations très voisines (exemple : «carte bleue» et «carte bleue visa»). Ces formes sont appelées formes variantes dont l'origine provient de l'expressivité du langage naturel. Dans la langue, chaque forme représente toujours un sens unique dans un contexte donné. Dans la description d'un domaine et pour une tâche précise, la granularité du sens autorise la réunion de plusieurs formes sans porter préjudice à la tâche fixée ; ce sera le cas de la catégorisation, dans l'exemple traité ici. Nous allons approfondir cette question des syntagmes dans la suite.

2.2. Corpus de textes

Dans le traitement du langage naturel nous pouvons opposer les «méthodes faibles» aux «méthodes exhaustives». Les premières sont issues des statistiques et les secondes proviennent de traitements locaux comme l'analyse grammaticale. D'après Teil & Latour, (1995), le traitement de corpus est plus adapté pour réaliser une acquisition de connaissances. En effet le corpus est bien adapté à l'exécution de méthodes statistiques (Basili, Pazienza & Velardi, 1997 ; Oakes, 1998). La linguistique de corpus a été développée depuis le début des années 1980. Comme les documents électroniques sont de plus en plus disponibles dans les institutions, les PME, les universités, les centres de données bibliographiques... tout comme les interviews d'expert, les corpus peuvent être créés et utilisés comme données d'entrée. Pour créer un corpus, deux problèmes sont à considérer : l'homogénéité et la taille. La taille est caractérisée par le nombre de mots. Un gros corpus doit avoir au moins 1 million de mots. Un corpus homogène couvre un domaine spécifique dans toute sa diversité. Un corpus est généralement écrit dans une langue bien qu'il puisse être multilingue. Certaines personnes prétendent que les corpus proviennent uniquement de collections de documentation technique parce que la terminologie y est bien établie, stable et nominalisée. En fait, comme les documents sont un vecteur de communication, s'ils reflètent un échange entre deux membres d'une communauté liés par un thème, nous pouvons considérer que la terminologie reflète aussi la terminologie de cette communauté. Nous décidons de focaliser notre étude sur de petits corpus (autour de 50 000 mots) en français concernant différents sujets (médecine, aéronautique, histoire...).

2.3. Concept

Nous avons basé notre définition du concept sur le sens et les relations (Tanguy & Thlivitis, 1996) auxquels l'approche par «frame» et la sémantique distributionnelle

peuvent être associées. Nous nous plaçons dans une définition plus proche de l'acquisition et de la formalisation de connaissance (Fisher, 1987 ; Michalski, 1980). Un concept se définit par des instances. Les instances pouvant être des termes ou des documents, de ce point de vue, la logique des prédicats est bien adaptée pour caractériser l'extension d'un concept (Kirsten, 1998). Un prédicat est un champ dont la valeur caractérise des objets pointant vers un concept, il peut en être ainsi avec des documents structurés (message électronique). Quand on veut lier les mots et les concepts, le problème devient plus difficile. Un mot simple pourrait être un concept aussi bien qu'un groupe de mots. L'influence du monde réel sur la combinaison logique du langage se matérialise dans les associations d'unités sémantiques de base (comme les termes). Par exemple, considérons les suites de termes : parc, fontaine, ombrelle. Les mots ne sont pas de même étymologie ni de même morphologie mais ils semblent appartenir au même champ sémantique : le jardin. Supposons qu'un homme des cavernes débarque dans notre monde moderne, il ne pourrait pas se représenter le sens de ce groupe du fait de son manque de culture. Nous ne pouvons nier que ce groupe représente un concept réel de notre représentation mentale. Les formes linguistiques ne sont pas suffisantes pour interpréter un cluster comme un concept. Un espace conceptuel a une relation avec l'espace des mots par des liens entrelacés avec l'espace pragmatique du monde réel. Traditionnellement on suppose qu'un concept se caractérise par une extension et une intension. L'intension est l'ensemble des propriétés d'un concept et l'extension est l'ensemble des instances matérialisant le concept. Dans les langages à objets, une classe est un ensemble d'objets possédant une structure, un comportement et des relations similaires. Nous baserons notre construction de concept à partir de cette description, mais a contrario de la plupart des méthodes courantes conduisant à une description polythétique :

- Si l'objet x est décrit par les attributs (a, b), l'objet y est décrit par les attributs (b, c) et l'objet z est décrit par les attributs (c, d), alors un cluster polythétique peut être :

$$K= \{(x, y, z) \in I^3 \mid d(x, y) < S \quad \text{et} \quad d(x, z) < S\}.$$

Nous proposons d'adopter une description monothétique d'un concept (voisine d'une classification par prototype), c'est-à-dire qu'un cluster a ses éléments proches d'un élément commun.

Par hypothèse, on se donne :

- I l'ensemble des objets à classer.
- $R = \{R_1, R_2, \ldots\}$ l'ensemble des règles statistico-linguistiques qui décrivent l'usage des objets à classer.
- $H = \{H_1, H_2, \ldots\}$ l'ensemble des heuristiques graphe-statistiques qui donnent le cadre du processus de «classification» et de généralisation des clusters par des étiquettes sémantiques.

On trouve une classification :

- Une suite $C_1, \ldots, C_n$ de sous-ensembles recouvrants et disjoints d'objets. On appelle cette suite l'ensemble K des clusters indicés par un entier $i \in N^*$.

$K = \{C_i = (a, b, c, \ldots)$ avec $i \in J$ et $\exists(a, b, c, \ldots) \in I \mid (a, b, c, \ldots)$ vérifient les contraintes de R et H$\}$ (espace de cliques agrégées indicées par des entiers de J) :

- Une description monothétique de clusters.
- Extraction dynamique.

En effet, on identifie deux sortes d'extraction de concept :

- Une extraction statique, C est défini par des attributs fixés avec des traits sémantiques
- Une extraction dynamique, C possède des termes typiques l'entourant.

Nous nous sommes plus spécialement focalisés sur le deuxième type d'extraction, mais un couplage entre ces deux sortes d'extraction serait préférable.

2.4. ONTOLOGIE

Depuis les cinq dernières années, des colloques, organisés par la communauté d'acquisition des connaissances, ont traité de la définition de ce type de structure qui pourrait contenir la connaissance d'un domaine. Nous observons qu'aucune structure standard n'a été développée et que la plupart de celles existantes sont spécifiques à un champ donné. Plusieurs tentatives proches du traitement du langage naturel ont conduit à des réseaux sémantiques de façon à stocker la connaissance de toute «l'humanité». Quelques projets, tels que Cyc qui a été dirigé par D. Lenat (1987), sont juste une tentative pour collecter et stocker l'information sans propriétés linguistiques. G. Miller (1986) a initialisé Wordnet, un autre projet typique mais sa structure n'est pas suffisamment générique pour supporter les parties du discours et les relations sémantiques d'un mot. Un projet similaire Semlex (Devin, 1998) a pour but de construire une structure de nœuds générique à deux couches : sémantique et lexicale. Quelques modèles prennent en compte la validation d'un expert à chaque étape de la construction d'une ontologie (Assadi, 1997). Un problème standard dans le traitement du langage naturel est la polysémie. Les méthodes de clustering ont prouvé leur efficacité pour obtenir un réseau de relations entre groupes de termes (Carpineto & Romano, 1996 ; Ibekwe-Sanjuan, 1996 ; Teil & Latour, 1995). Décrire un réseau par des relations spécifiques comme l'hyperonymie peut aussi générer du bruit. Considérons deux termes T_1 (avion) et T_2 (bateau), T_1 a les traits sémantiques F_1 (air) et F_2 (moteur), et T_2 a les traits sémantiques F_3 (mer) et F_2 (moteur). Ils sont liés au même hyperonyme H (véhicule) décrit par F_1, F_2 et F_3. T_3 (planeur) peut être synonyme de T_1 par le trait sémantique F_1 (air). Une action utilisant T_1 pourrait être généralisée en utilisant H mais comme H implique T_2 et comme T_2 et T_3 sont équivalents ce qui signifie que T_1 et T_3 seraient utilisés ensemble sans trait sémantique en commun. Une relation plus puissante serait la relation a-pour-partie ou l'holonymie. Un holonyme pourrait être une bonne généralisation non-polysémique d'instances. Deux autres relations fondamentales utiles seraient la synonymie et la référence. Ces trois relations appartiennent au clustering par convergence. Ainsi un algorithme de clustering efficace produirait des instances liées les unes aux autres par ces types de relations : holonymie, synonymie et référence, mais sans les qualifier au sein du cluster. Le lien hyperonymique pourrait n'être utilisé que par contrainte. La première génération d'ontologie a modélisé l'ontologie à partir de rien, et maintenant l'espace d'information (textuel) aide à construire une ontologie (Feng, Copeck, Szpakowicz & Matwin, 1994 ; Maikevich, 1998). L'ontologie se concevrait comme un modèle dirigé par clustering, ce dernier servant à qualifier quelques liens. Le processus de clustering serait un outil pour le remplissage de l'ontologie. En d'autres termes, l'alimentation de l'ontologie serait le système de représentation des connaissances (SRC) et les données clusterisées rempliraient le SRC en faisant des corrections pour

qualifier les liens ; le système dans sa globalité est l'ontologie associée à la base de données textuelles.

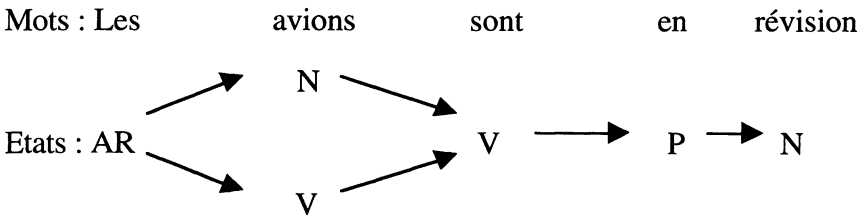
3. EXTRACTION DE TERMES

3.1. METHODES D'EXTRACTION

Un terme est construit par deux composants : une tête et une expansion. Quatre méthodes peuvent être implémentées pour extraire des termes d'un texte. La première méthode, statistique, est appelée «méthode des segments répétés». Cette méthode exploite un corpus ainsi que l'espace des fréquences des mots pour collecter les têtes dans un premier temps. La deuxième étape consiste à extraire l'expansion des têtes précédentes. Le processus est itératif jusqu'à ce qu'aucune expansion ne soit plus détectée. La seconde méthode est la méthode des bornes. On utilise un dictionnaire de bornes et le but consiste à extraire une expression entre deux bornes. Le dictionnaire peut être un ensemble de mots ou un ensemble d'étiquettes grammaticales. Le but est de détecter les séquences les plus fréquentes de mots dans un corpus. Des scores de collocabilité montrent la force de liaison syntaxique entre paires ou séquences de mots. La dernière méthode est une méthode de reconnaissance de patron appelée «méthode de patron morphosyntaxique». Cette méthode peut être divisée en deux types : statistique ou Chaîne de Markov Cachée (Hidden Markov Model ou HMM), et une méthode basée sur des règles. Nous utilisons la méthode HMM pour extraire les termes de corpus. Premièrement, le texte est étiqueté en utilisant un ensemble d'étiquettes grammaticales prédéfinies. Ensuite, le modèle d'apprentissage HMM aide à «désambiguïser» les phrases (i.e. à leur donner un sens univoque). Finalement l'activation de patrons de reconnaissance permet d'extraire des syntagmes nominaux ciblés (Schiller, 1996).

3.2. MODELE DE MARKOV CACHE

Dans les modèles de Markov observables, chaque état est équivalent à un état observable. L'état présent dépend seulement de l'état précédent. Dans notre cas, une étiquette grammaticale (verbe, adverbe, nom...) représente un état. Un HMM aide à étiqueter les phrases et résoudre les ambiguïtés. Une ambiguïté est modélisée par différents chemins dans un graphe ou une suite de séquences d'états (Chanod & Tapanainen, 1995).



A : adjectif ; N : nom ; V : verbe ; AR : article ; P : préposition

Cependant, pour un HMM, la sortie n'est pas la séquence d'état interne, mais une fonction probabiliste de cette séquence interne. Quelques personnes font des symboles de sortie d'un modèle de Markov une fonction de chaque état interne, tandis que d'autres font de la sortie une fonction des transitions. À chaque état donné, il y a un choix de symboles, chacun avec une certaine probabilité d'être sélectionné. Une chaîne de Markov

cachée est un processus doublement stochastique. Elle consiste en un processus stochastique qui ne peut pas être observé, décrit par les probabilités de transition entre paires d'états et calculées à partir d'un corpus d'apprentissage. Deuxièmement, un processus stochastique gère les symboles de sortie qui peuvent être observés à partir des données d'entrée au processus, et représentés par les probabilités de sortie du système. Les principaux paramètres du HMM peuvent être résumés par l'ensemble des probabilités de transition, l'ensemble des probabilités de sortie et l'état initial du modèle.

L'utilisation de modèles de Markov cachés dans la résolution d'un problème d'étiquetage implique trois problèmes algorithmiques : l'apprentissage, l'évaluation, l'estimation. Pendant l'apprentissage, les paramètres initiaux du modèle sont ajustés et maximisés afin d'observer une séquence de symboles. Cela rendra le modèle actif pour prédire de futures séquences de symboles. L'apprentissage implémente l'algorithme de re-estimation de Baum-Welch [http 1]. À ce stade un corpus étiqueté à la main permet de calculer les paramètres du modèle (probabilité de transition d'être dans l'état i à la position p et de suivre un état j à la position p+1). Le problème de l'évaluation est celui de calculer la probabilité qu'une séquence observée de symboles apparaisse comme résultat d'un modèle donné. Il est résolu en utilisant un algorithme forward-backward [http 2]. Dans le problème d'estimation, nous observons une séquence de symboles produite par une chaîne de Markov cachée. Il s'agit d'estimer la séquence d'états la plus probable que le modèle permet d'obtenir pour produire cette séquence de symbole. L'algorithme de Viterbi permet de calculer une telle séquence [http 3].

3.3. Extraction de patrons

Après avoir obtenu un corpus proprement étiqueté et «désambiguïsé», l'étape suivante d'extraction des termes consiste en quelques règles de grammaire pour extraire des groupes nominaux (GN). Certaines études sur des corpus français montrent que les groupes nominaux fréquents apparaissant dans les textes sont seulement au nombre de 4 : Nom-Adjectif, Adjectif-Nom, Nom-Préposition-Nom et Nom-Nom. Ces séquences représentent plus que 70 % des syntagmes nominaux dans les textes. Bien entendu il est possible d'enrichir la grammaire en ajoutant un adverbe ou un adjectif dans un groupe nominal pour obtenir des formes variantes comme Nom-Adjectif-Préposition-Nom ou Nom-Adverbe-Adjectif... etc. Cette action étend la collecte des GN intéressants en réduisant des formes variantes à leur forme la plus fréquente. Cette réduction est particulièrement importante pour une approche par clustering que nous verrons au paragraphe 4. Les règles de grammaire sont spécifiées en tant qu'expressions régulières récursives. Le texte d'entrée est transformé en un texte de sortie étiqueté traité par un compilateur d'expressions régulières. Toutes les étapes du traitement (étiquetage et extraction des GN) sont implémentées grâce à des automates à états finis. Cela signifie que les chaînes de caractères sont converties en arbre avec des nœuds simples et des nœuds finaux. Les automates à états finis sont largement utilisés par les compilateurs de langage. Fortement optimisés, ils permettent d'accélérer les temps de traitement et peuvent aussi réduire le stockage de données (dictionnaire...).

4. LA MÉTHODE DE CLUSTERING

Notre module de classification peut être divisé en 5 sous-modules (Figure 1). Un compilateur de corpus donne un fichier position (1). Ce fichier contient chaque forme lemmatisée avec toutes ses positions par rapport au premier mot du corpus. L'extracteur

de termes donne des fichiers de termes (2) comme fichier d'entrée au constructeur de matrice et à l'extracteur de termes pôles.

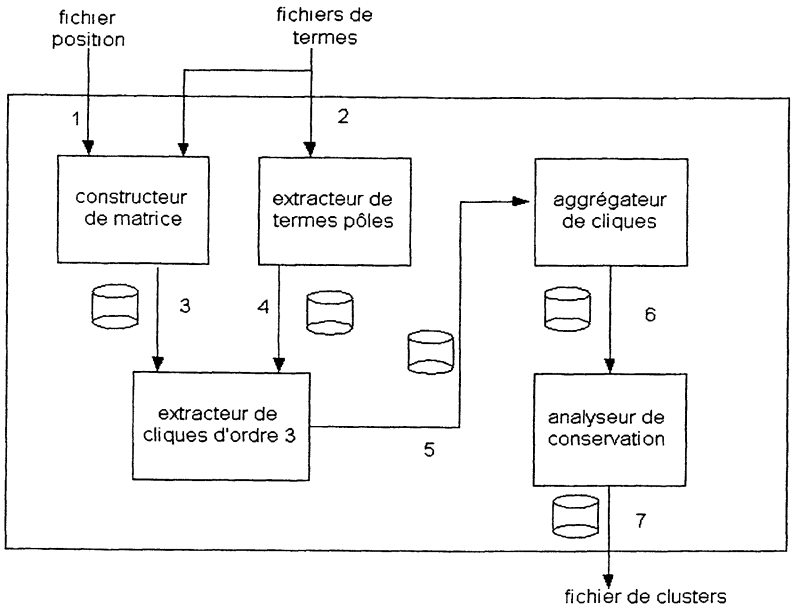


Figure 1. Vue générale du module de classification

La sortie du constructeur de matrice est une matrice de co-occurrence (3). La sortie de l'extracteur de termes pôles est un fichier de termes pôles (4). Un extracteur de clique d'ordre 3 utilise les deux précédents fichiers comme entrée pour donner un fichier de cliques d'ordre 3 (5). Ce fichier est utilisé comme entrée par un agrégateur de cliques qui produit un fichier de clusters (6). Il est utilisé comme entrée par un analyseur de conservation qui le complète (7). Finalement ce dernier fichier est utilisé par un gestionnaire de thésaurus pour définir une structure hiérarchique de clusters par thèmes.

4.1. TABLE DE CONTINGENCE

Les tables de contingence ont été utilisées depuis longtemps avec les bases de données relationnelles pour la découverte de taxinomies et de régularités (Zytkow & Zembowicz, 1997) montrent que ce type de table est une structure de données idéale pour assurer un support de découverte de connaissances et la corrélation entre événements conduisant à des régularités. Le format est même bien adapté pour exploiter les relations entre données symboliques. Nous basons notre première étape de la méthode sur la création de cette table. Dans notre méthode, la table est traitée comme une matrice M représentée par son coefficient général m_{ij}. Contrairement à une table relationnelle standard variable/attribut, nous n'avons aucune description des objets selon leurs propriétés. Nous remplissons la matrice avec les associations du terme i (variable) et du terme j fixant le coefficient m_{ij}. Une fenêtre de mots caractérise l'association syntaxique valide. Quand une association existe entre deux items nous l'appelons co-occurrence (ou collocation) (Mikheev & Finch, 1992 ; Smadja & McKeown, 1990). Une co-occurrence est évaluée en fonction du texte source initial et non d'après la structure morphologique d'un terme comme nous pouvons le voir dans certains modèles de classification (Assadi, 1997). Les hypothèses développées tiennent compte de la structure d'un terme sous forme d'une tête et d'une expansion (par exemple dans «château fort», «château» est la tête et «fort» l'expansion). La démarche consiste à raisonner sur le nombre de têtes ou d'expansions communes pour

créer des classes distinctes. Nous considérons que le corpus n'est pas forcément riche seulement en termes de têtes et expansions communes. Deuxièmement, ce type de corrélation devrait concerner la structure morphologique plus qu'une force de lien sémantique entre items. Une co-occurrence est un état d'association utilisé dès la naissance de l'informatique pour accéder au contexte. Il en résulta une forme primitive de système appelé concordancier. Un concordancier est un outil permettant de rassembler les contextes syntaxiques d'un terme dans un texte. Dans certains cas il peut collecter tous les bigrams d'un texte. Smadja & McKeown (1990) montrent que les co-occurrences proposent une définition d'un concept n'étant pas spécifié dans un dictionnaire. Cela correspond assez bien avec notre définition dynamique du concept grâce au contexte. Le Tableau 2 nous présente un cas correct de distribution de données pouvant amener une méthode standard de clustering à un résultat convenable. On obtiendrait les clusters (1,2,3,4) and (5,6,7) sans recouvrement.

	1	2	3	4	5	6	7	8	9
1	*	*	*	*					
2		*	*	*					
3			*	*					
4				*	*	*	*		
5					*	*	*		
6						*	*		
7							*		
8								*	
9									*

Tableau 2. Cas idéal de matrice de co-occurrence.
(* corrélations présentes non bruitées)

	1	2	3	4	5	6	7	8	9
1	*	*	...	*	#				#
2		*	...	*		#			#
3			*	*	#				
4				*	*	...	*		
5					*	*	...	#	*
6						*	*		#
7							*	#	#
8								*	#
9									*

Tableau 3. Cas réel de matrice de co-occurrence.
(* corrélations présentes non bruitées, # corrélations présentes bruitées,
... corrélations manquantes)

Malheureusement comme on peut le constater sur le tableau 3, après un traitement de classification standard, les données ont encore des irrégularités accusant des valeurs manquantes et des valeurs bruitées. Les résultats seraient (1,2), (3,4,5) et (7,8,9). La transition d'un cas idéal vers un cas réel subdivise le premier cluster idéal et détruit le second cluster idéal. Nous voyons que les items 4 et 9 doivent être aussi associés au

cluster (1,2). Grâce aux associations transversales, telles qu'elles pourraient être détectées avec notre approche par analyse de graphe, la corrélation précédente serait satisfaite.

4.2. REDUCTION CANONIQUE DE TERMES

Comme le Tableau 2 le montre, nous devons trouver les valeurs manquantes pour aboutir à des blocs homogènes. Une solution possible consiste à synthétiser les formes en formes canoniques. La consistance des co-occurrences pourrait s'atténuer à cause de la variété des formes. Durant des siècles, les langues ont su créer des familles morphologiques de mots et d'expressions avec approximativement le même sens. Pour nous, le phénomène linguistique n'est pas négligeable. Ce phénomène linguistique est partiellement traité dans les produits du marché en recherche d'information et connu sous le nom de stemming. Pour l'exploiter, nous avons besoin d'appliquer deux types de connaissance linguistique. La première est l'équivalence entre les formes usuelles et leur lemme associé. Nous appelons l'action utilisant cette première connaissance : lemmatisation. Cette première connaissance doit être appliquée aux mots courants du fait de leur forme irrégulière. En effet, les mots utilisés dans le dialogue et les documents écrits ont un comportement morphologique irrégulier particulièrement en français par opposition aux mots nouveaux qui suivent des règles de construction restreintes. La seconde connaissance est une liste de suffixes standard. Elle sera utilisée pour les mots spécifiques provenant d'un domaine technique ou d'un jargon. Nous appelons l'action utilisant cette connaissance : troncature. Ainsi ces deux actions sont effectives sur les mots simples : lemmatisation et troncature. Mais ces deux processus concernent seulement les variations de monotermes et non de variation de multitermes. Un autre phénomène linguistique complexe apparaît avec les formes variantes composées (Polanco, Grivel & Royauté, 1995). Les groupes nominaux composés ou multitermes peuvent être déclinés en différentes structures ayant des similitudes sémantiques tels que «accélération d'un électron libre» et «accélération d'un électron». Nous distinguons trois principales variations : insertion, expansion et permutation. Ces variations prennent leur origine dans des propriétés géométriques mais pour certaines, comme la variation par permutation, les facteurs sémantiques sont utiles ; par exemple pour corréler «électron accéléré» et «accélération d'un électron» en rapprochant le verbe «accélérer» et le nom «accélération» dans une même famille sémantique. Une des variations les plus simples à traiter est l'insertion. Notre hypothèse de base est la suivante : dans une langue, deux formes différentes expriment un sens différent, même si la différence est faible, mais certaines expressions sont plus fortement corrélées par leur sens que les autres. Malheureusement les théories de la linguistique moderne ne nous apportent pas de formalisme pour différencier quantitativement deux termes donnés par rapport à des traits sémantiques préalablement fixés.

4.3. ECHANTILLONS DE TERMES

Pour établir notre méthode de clustering, dans un premier temps nous sélectionnons les termes les plus pertinents du fichier de sortie donné par l'extracteur de groupes nominaux. L'extracteur de GN nous donne une liste non triée de groupes nominaux trouvés dans le corpus. Un tel résultat n'est pas directement exploitable. Nous soumettons deux contraintes pour obtenir une entrée adaptée à notre système. La première contrainte est le filtrage de fréquence. Les fréquences sont le nombre d'occurrences d'un groupe nominal dans un corpus. Nous choisissons 2 comme seuil de filtrage. Ainsi nous obtenons l'équivalent de segments répétés sur la base des fréquences des chaînes de caractères. Nous pensons que les expressions fréquentes sont plus représentatives de la terminologie du domaine que les expressions non fréquentes. Nous devons prendre garde que les

expressions fréquentes ne sont pas majoritaires dans un corpus. Ainsi la quantité d'information résultante n'est pas susceptible de montrer des corrélations d'information non détectables par lecture séquentielle d'un document. Mais, en utilisant des méthodes statistiques et, comme nous l'avons expliqué au paragraphe 2.1, nous décidons de traiter les corpus avec des méthodes faibles pour gagner en robustesse. On appelle hapax un mot qui a une fréquence unité, c'est-à-dire qui n'apparaît qu'une seule fois dans le corpus. La proportion d'hapax dans un corpus est souvent supérieure à 60 %. Pour palier à ce déficit d'information inutilisée, quoique très bruitée, nous filtrons les termes par une série d'heuristiques de sélection sur des critères statistiques. Ces critères doivent également éviter de rendre la méthode de clustering exponentielle en temps de calcul. Nous ne disposons pas de critère donnant une mesure de la couverture du domaine par les termes sélectionnés. À défaut d'avoir une métrique qui quantifie cette couverture ou d'avoir le commentaire d'un expert du domaine, nous supposons que les termes couvrent suffisamment le domaine décrit par le corpus. Le second paramètre de filtrage permet d'obtenir le fichier de termes définitifs. Il s'agit d'un paramètre de discrimination. En fait le paramètre est double : il concerne la structure du corpus avec les documents et avec les paragraphes. Nous définissons un corpus comme une collection de documents séparés. Nous définissons un paragraphe comme une unité textuelle séparée d'une autre par un saut de ligne multiple ou un couple d'astérisques (placé à la main pour les tests) et un saut de ligne. Le paramètre de discrimination par paragraphe s'écrit $D_p = Nw_p/Nt_p$ où Nw_p est le nombre de paragraphes contenant le mot, Nt_p est le nombre total de paragraphes dans un corpus. Le paramètre de discrimination par document $D_d = Nw_d/Nt_d$ où Nw_d est le nombre de documents contenant le mot, Nt_d est le nombre total de documents dans le corpus. Nous utilisons plus couramment le paramètre de discrimination (D_p) en coupant la sélection au seuil de 0.03. Le second échantillon approprié dans notre méthode est un fichier de tous les verbes exprimés dans le corpus. Les verbes sont essentiellement communs et bien répertoriés dans les dictionnaires avec leurs flexions. Nous pouvons facilement les détecter dans un corpus et les stocker dans un fichier spécifique. La troisième étape de création de l'échantillon de termes est très importante et consiste à sélectionner un sous-échantillon du fichier de termes. Nous appelons les éléments de cet échantillon les termes pôles. Nous avons conduit une étude empirique sur un corpus médical nous ayant amené à construire des clusters à la main sur la base du contenu médical conceptuel. Les résultats nous ont permis d'observer une répartition des termes de chacun des clusters autour d'un terme spécifique dont la fréquence est médium par rapport à l'étendue des fréquences de tous les termes. Cela correspond à notre idée de construire des clusters avec une structure monothétique. Après l'étape du préclustering nous rentrons dans le cœur du processus.

4.4. Utilisation de schemas linguistiques

Nous dégageons notre approche de la voie structuraliste de description du langage. Une recherche de fouille dans un corpus peut révéler des relations non-aléatoires (Harris, 1968 ; Habert, Naulleau & Nazarenko, 1996). Quelques relations peuvent être appelées «schéma» du fait de leur composition. Nous nous intéressons notamment aux structures relationnelles de schéma verbe-GN. D'autres types de schémas pourraient servir dans le repérage de relations mais nous disposons d'un fichier de verbes, et donc le schéma verbe-GN s'impose naturellement pour croiser les co-occurrences dans un traitement matriciel. Nous pouvons espérer que les verbes spécifiques soient utilisés syntaxiquement devant/après un élément d'une terminologie (Rousselot & Frath, 1996). Ce n'est pas ce que nous observons. Mais comme les verbes représentent une typologie d'état et d'action, ils impliquent une utilisation spécifique d'attributs. Nous exploitons le rôle des verbes

comme marqueurs des relations entre GN. La linguistique computationnelle pure permettrait de trouver des schémas typiques de la forme [terme A][verbe V][terme B] plusieurs fois. Ainsi une règle d'inférence permettrait de grouper le terme B et le terme C de par leur relation [terme A][verbe V][terme C]. Dans notre méthode d'analyse de données, nous compilons toutes les relations verbales liant un terme A et un terme B. Ces relations seront mises en évidence par transposition de la matrice de co-occurrences. Des corrélations similaires ont été développées en informatique documentaire pour exprimer des relations entre termes et documents. Une matrice terme*documents est construite et transposée pour obtenir des ensemble lexicaux.

4.5. Recherche de cliques

J.L Kuhns en 1959 déclare que le clustering basé sur des graphes peut être bénéfique à l'indexation terminologique. Mais à cette époque aucune application informatique n'a validé cette hypothèse. Sparck-Jones, (1964) ; Augustson & Minker, (1970) ont optimisé des algorithmes de recherche de clique pour l'appliquer à une matrice terme*document. Ils ont pu extraire un certain nombre de clusters intéressants à partir d'un ensemble de 4 000 termes. Comme nous le savons, l'extraction de sous-graphes à partir d'un graphe est un problème NP-complet. C'est pourquoi depuis les années 70 aucune application n'a vraiment utilisé l'extraction de graphes de façon significative. Nous pensons que le clustering par graphe pourrait répondre à notre postulat du fait qu'il s'implémente par association, et que les liens entre variables sont traités séparément. La recherche de cliques n'est qu'une étape dans notre modèle de clustering par heuristiques. Cette recherche est de plus incrémentale dont l'ordre, défini plus loin, des cliques recherchées évolue.

Soit l'ensemble d'items I dénotant l'ensemble des vertex ou sommets des graphes (les termes dans notre cas). Un hypergraphe sur I est une famille $H = \{E_1, E_2, \ldots, E_n\}$ d'arêtes ou sous-ensembles de I, tels que $E_i \neq \emptyset$ et $\bigcup_{i=1}^{n} E_i = I$. Un hypergraphe simple est un hypergraphe tel que, $E_i \subset E_j \Rightarrow i = j$. Un graphe simple est un hypergraphe simple pour lequel les arêtes ont une cardinalité 2. La matrice de co-occurrence coïncide avec la matrice d'incidence sur un graphe simple. La cardinalité maximum d'arête est appelé le rang, $r(H) = \max_j |E_j|$. Si toutes les arêtes ont la même cardinalité, alors H est appelé hypergraphe uniforme. Un hypergraphe uniforme simple de rang r est appelé hypergraphe r-uniforme. Pour un sous-ensemble $X \subset I$, le sous-hypergraphe induit par X est ainsi défini, $H_X = \{E_j \cap X \neq \emptyset \mid 1 \leq j \leq n\}$. Un hypergraphe complet r-uniforme avec m sommets, dénoté par K^r_m, consiste en tous les r sous-ensembles de I. Un sous-hypergraphe complet r-uniforme est appelé clique d'hypergraphe r-uniforme. Une clique d'hypergraphe est maximale si elle n'est contenue dans aucune autre clique. Pour les hypergraphes de rang 2, cela correspond au concept familier de clique maximale dans un graphe. Dans la prochaine partie de l'article, nous appelons une clique un sous-hypergraphe maximal complet 2-uniforme (i.e une graphe dont tous les sommets sont reliés entre eux). Nous définissons l'ordre o d'une clique C comme la cardinalité de son ensemble d'arêtes N distinctes, o = card(N (C)). La première étape que nous mettons en œuvre est de collecter toutes les C avec o = 3 :

$$K_3 = \{C = i, j, l) \text{ avec } i \in P \text{ et } j, l \in I = (1, \ldots, n) \mid o = 3\}$$

où P est l'ensemble des termes pôles.

DÉFINITION : Soit freq_max le max de la fréquence d'un terme du fichier d'individus. Un terme est considéré terme pôle si sa fréquence est entre les bornes min*freq_max et max*freq_max avec min et max compris entre 0 et 1 exclus.

Cela correspond à une heuristique étudiée avec les classes de termes médicaux.

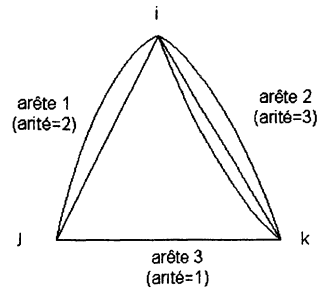


Figure 2. Exemple de clique d'ordre 3

Nous avons trouvé des liens de co-occurrence entre les éléments de clusters «idéaux» établis à la main. Les résultats montrent qu'un terme pôle attire de bonnes co-occurrences et possède une fréquence dans un certain intervalle. Cette configuration basée sur une heuristique modélise notre structure monothétique de cluster.

4.6. AGREGATION DE CLIQUES

Lors de la troisième étape du processus de clustering nous utilisons une heuristique d'association pour clusteriser ensemble des sous-graphes. Pour chaque clique d'ordre 3 nous réalisons le prolongement pour former des cliques d'ordre 4. Nous regroupons trois cliques d'ordre 3 qui ont le même terme pôle indépendamment de la position des sommets. Nous obtenons l'ensemble :

$$K_4 = \{C = (i, j, l, m) \quad \text{avec} \quad i \in P \quad \text{et} \quad j, l, m \in I = (1, \ldots, n) \mid o = 4\}.$$

Ensuite, la quatrième étape du processus consiste en l'union de plusieurs cliques d'ordre 4 de façon à former des clusters.

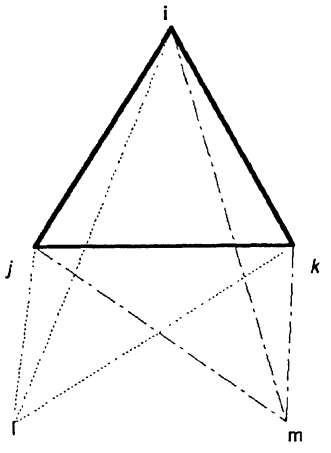


Figure 3. Agrégation de cliques d'ordre 4.

Cette phase nécessite deux conditions pour être opérationnelle. La première est d'avoir le même terme pôle dans chaque clique d'ordre 4 agrégée. La seconde condition est d'avoir le même couple de termes, que nous appelons termes pivots, dans chaque clique d'ordre 4. Le triplet (terme pôle, terme pivot 1, terme pivot 2) est très proche de nos hypothèses et participe à notre construction monothétique du cluster. Soit sous forme symbolique la clique agrégée K_{agg} :

$$K_{agg} = \{ \cup_{k=1}^{\mu} C_k \text{ avec } C_k = (i, j, l, x_l) \text{ et } C_{k'} = (i, j, l, x_{k'}) \; \forall k, k' \in \{1,...,\mu\} \text{et } C_k, C_{k'} \in K_4 \}$$

où μ est le nombre total de cliques d'ordre 4 ayant les mêmes triplets de termes pôle et pivot.

4.7. HEURISTIQUE DE QUASI-CONSERVATION

Nous ne pouvons pas décrire la distribution des mots avec des distributions classiques (gaussienne, poissonienne...) parce que les écart-types de la distribution des mots sont très grands. Les distributions sont étalées. En général quand les distributions sont irrégulières, le rang de classement est utilisé pour comparer des échantillons entre eux. Zipf (1935) a étudié la distribution des mots avec leur rang et a montré l'existence d'une loi de puissance liant le rang et la fréquence des mots. La loi, typique du domaine du langage naturel, semble être très bien observée dans la nature comme une loi de puissance (Kanter & Kessler, 1995). Ainsi pour comparer des échantillons nous utilisons la variation de distribution plutôt qu'un test paramétrique tabulé standard. Des tests non-paramétriques pourraient donner des résultats insatisfaisants difficiles à détecter automatiquement (Edmonds, 1997 ; Yarowsky, 1992). Notre heuristique se veut équivalente à celle d'un test d'adéquation implémenté en quatre étapes. La première étape consiste à comparer tous les éléments d'un fichier de termes avec un terme pôle (T_p). Nous gardons les termes qui ont un coefficient matriciel plus grand qu'un certain seuil (Tableau 4). C'est la première condition permettant de sélectionner un candidat terme (T_c). La seconde étape consiste à faire coïncider toutes les relations de co-occurrences T_c et T_p obtenues avec les hapax. Un hapax est un mot apparaissant une fois dans le corpus. Nous vérifions si le nombre d'hapax liés ensemble par co-occurrence est plus grand qu'un certain seuil. Cette seconde condition valide la deuxième étape.

	Hapax 1	Hapax 2	...	Hapax n
T_c	A	b	...	C
T_p	a'	b'	...	c'

Tableau 4. Distribution des hapax

La troisième étape consiste à rassembler toutes les relations de co-occurrence de T_c et T_p obtenues avec les non-hapax (Tableau 5).

	Word 1	Word 2	...	Word n
T_c	aa	bb	...	cc
T_p	aa'	bb'	...	cc'

Tableau 5. Distribution de non-hapax

- Soit f la distribution pour le terme cible : $f_1 = aa$ $f_2 = bb$ ... $f_n = cc$ et $S_f = \Sigma f_i$
- Soit g la distribution pour le terme pôle : $g_1 = aa'$ $g_2 = bb'$ ... $g_n = cc'$ et $S_g = \Sigma g_i$

Dans la dernière étape nous estimons une fonction de déviation. Nous coupons à 10 % des fréquences les plus faibles et 10 % des fréquences les plus élevées des mots trouvés à l'étape précédente. Nous appelons cette fonction de déviation ε. Elle évalue la différence distributionnelle entre les mots fréquents dans le voisinage de T_p et de celui de T_c. Si les deux distributions sont équivalentes (i.e. proportionnelles) alors ε tend vers 0.

Nous modélisons la distribution observée de T_c par $f(x) = 1/1+x$.

La même distribution observée pour T_p par $g(x) = \alpha/1+x$. (α est une constante).

Nous explicitons la quasi-conservation par la relation $\int f(x)\,dx = \alpha \int g(x)\,dx$.

Ainsi $(f'(x) \int g(x)\,dx - g'(x) \int f(x)\,dx) / (\int g(x)\,dx)^2 = 0$ dans le cas absolu.

Dans notre cas $(f'(x) \int g(x)\,dx - g'(x) \int f(x)\,dx) / (\int g(x)\,dx)^2 = \varepsilon$

M(i,j) : coefficient matriciel de co-occurrence.

S_g : somme de M(i, j) pour T_p $\int g(x)\,dx$.

S_f : somme de M(i, j) pour T_c $\int f(x)\,dx$.

Δ_g : différence entre M(i, j) et M(i, j-1) pour T_p.

Δ_f : différence entre M(i, j) et M(i, j-1) pour T_c .

$$\varepsilon = \left| \frac{\sum_{i=2}^{p_{max}-1} \left(S_g \cdot \left| \Delta f_{i-1}^{i+1} \right| - S_f \cdot \left| \Delta g_{i-1}^{i+1} \right| \right)}{2 S_g^2} \right|$$

Les clusters que nous obtenons en définitive à l'issue de cette dernière heuristique de quasi-conservation sont :

$$K = \{C_j = C_k \cup (m_1,...,m_\mu) \text{ avec } C_k \in K_{agg}$$
$$\text{et } i \in P \text{ et } i \in C_k \mid C_j \cap C_{j'} \leq 2 \text{ et } G(i) \approx F(m_1,...,m_\mu)\}$$

où G et F sont des distributions zipfiennes approchées.

4.8. Generalisation des classes

Une fois que l'on obtient des clusters finaux, la dernière étape concerne la généralisation des instances et la construction d'un arbre hiérarchique de clusters recouvrants. Il a été montré que la généralisation aide à l'interprétation. Quand on l'applique aux clusters la généralisation donne des résultats satisfaisants (Capponi & Toussaint, 1998). Tous les termes pôles du fichier de sortie sont groupés par terme pôle. En effet un terme pôle peut générer deux ou plusieurs clusters. Chaque terme pôle est identifié par une étiquette numérotée. Pour chaque cluster nous affichons un terme pôle et les termes pivots

associés. Un lien relie le terme pôle au panneau des instances qui y sont rattachées. Nous utilisons un thésaurus pour étiqueter les clusters. Son action est dédiée à l'étiquetage de cluster en deux niveaux. Il permet la recherche de thèmes pertinents de clusters et de superclusters.

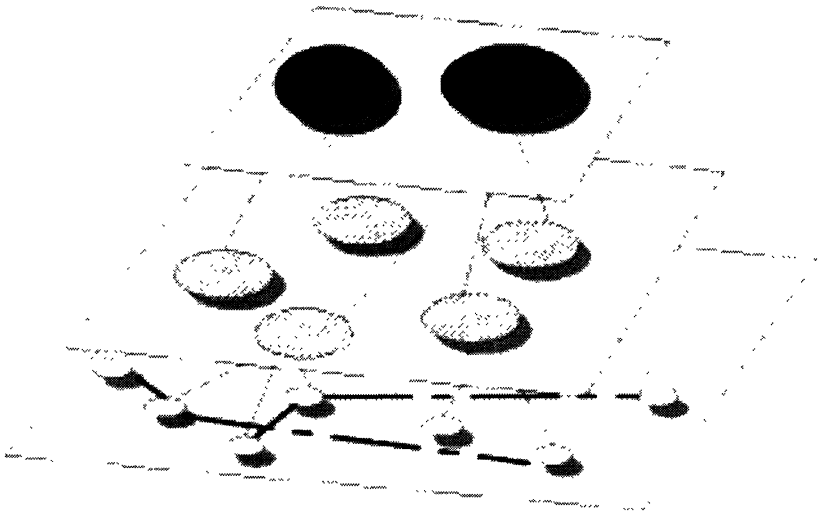


Figure 4. Relations de l'ensemble pré-structuré de concepts

Il permet aussi de retourner deux ou trois thèmes prédominants du corpus s'ils existent. Il arrive que certains clusters ne soient pas reconnus par la liste des thèmes issue de la structure du thésaurus. Pour chaque cluster, un panneau présente des cases à cocher pour le cluster entier et pour chaque instance d'un cluster. Ces cases à cocher procèdent à une sélection des éléments les plus intéressants d'un cluster. Un lien permet un retour à la liste des termes pôles pour localiser un cluster. Un utilisateur peut vérifier certaines expressions et les retenir. L'information est envoyée à un programme CGI (Common Gateway Interface) dont le rôle est de corriger le fichier de clusters. Les modifications sont prises en compte seulement si un utilisateur les valide.

PROPRIÉTÉ. L'élimination d'un terme pivot ou d'un terme pôle détruit le cluster. L'élimination d'un terme d'une autre nature conserve le cluster sans le terme.

Une instance peut appartenir à plusieurs clusters. Ainsi les liens hypertextuels assurent une navigation d'un terme vers les clusters auxquels il appartient. Un terme est un nœud le reliant à un panneau affichant les termes pôles et les termes pivots concernant les clusters affiliés. Un gestionnaire de thésaurus structure les clusters en créant une hiérarchie.

Le thésaurus est divisé en 3 parties :

– 1ère partie : *Liste de catégories*. Deux couches de catégories constituent la structure de la hiérarchie. Pour chaque catégorie du niveau inférieur (900) un code lui est associé. Les intervalles des codes précédemment cités sont associés à des catégories de niveau supérieur (30).

– 2ème partie : *Corps des catégories*. Cette partie est plus irrégulière que les deux autres. Pour une catégorie donnée, les termes de la famille sémantique sont groupés en catégories grammaticales (adjectifs, noms, adverbes, verbes). Différents sous-thèmes sémantiques

sont séparés par plusieurs paragraphes. Certaines informations sont écrites entre parenthèses. Un terme peut être séparé d'un autre par un point, un tiret ou un point-virgule. Un terme peut avoir un code pointant vers une autre catégorie. Un signe peut précéder une locution (comme 'fam. :', 'ou', 'mus.',...).

– 3ème partie *: Index.* Il donne un fichier inversé de la deuxième partie. Un terme pointe vers tous les codes de catégories dans lesquelles il est inclus.

Maintenant nous présentons un algorithme basé sur des heuristiques implémentant une généralisation de clusters. Un lien de généralisation est similaire à un lien sémantique ; il est aussi appelé lien hyperonymique. Le processus génère deux couches de généralisation. À la première étape nous cherchons les codes communs à tous ou au moins deux instances d'un cluster. Si certains codes sont en même proportion, nous choisissons celui représentant une catégorie entière (en gras) ou sinon nous choisissons le plus petit. Si un tel code existe, il devient l'étiquette sémantique du cluster ou sinon nous cherchons le plus petit code du terme pôle en le sélectionnant comme étiquette sémantique. Dans le cas où le terme est composé nous cherchons le code du terme entier ou sinon tous les codes du premier mot et du dernier mot du terme seulement s'ils sont de catégorie nominale. Dans la deuxième étape nous rattachons un code de catégorie de niveau supérieur à chaque code lié à un cluster dont la valeur se trouve dans l'intervalle du code de niveau supérieur. Par exemple, le code 248 est dans l'intervalle 230-267 correspondant à «matière» ; ainsi le nœud de niveau supérieur est «matière».

Nous sélectionnons les thèmes prédominants grâce à l'heuristique suivante : nous trions tous les codes représentant les instances des clusters dans un ordre décroissant. Nous sélectionnons les trois premiers codes ayant une fréquence minimale de 3. Si le premier code a une fréquence inférieure à 3 alors aucune catégorie n'est sélectionnée. Si plus de trois catégories apparaissent alors nous choisissons les trois qui ont le code de catégorie le plus faible. Dans le prochain chapitre, un exemple sera donné.

4.9. INCREMENTALITE

Nous ne pouvons pas espérer résoudre une extraction de graphe en temps CPU linéaire. Le clustering par graphe est consommateur de temps de calcul. L'incrémentalité des processus autorise le traitement de grandes quantités de données, i.e de gros corpus ou des milliers de termes dans notre cas (Fisher & Schlimmer, 1997 ; Thomson & Langley, 1988). Dans la méthode, le nombre de cliques d'ordre 3 et de cliques d'ordre 4 est proportionnel au nombre de termes et de coefficients non nuls de la matrice. Plus nous recueillons d'associations et plus le nombre de sous-graphes augmente. Pour un fichier de 1 000 termes, nous pouvons facilement atteindre plusieurs milliers de cliques d'ordre 3. La solution est de stocker les résultats temporaires dans une base de données. Cinq tables sont nécessaires : une table pour la matrice, une table pour les termes pôles, une table pour les cliques d'ordre 3, une table pour les cliques d'ordre 4 et une table pour les clusters. Les étapes de l'incrémentalité sont les suivantes :

1. Le stockage d'une nouvelle ligne/colonne de la matrice si un nouveau terme apparaît.
2. Le re-calcul de la fréquence maximum et si elle change le re-calcul des fréquences bornes pour l'extraction des termes pôles et le stockage des nouveaux termes pôles ou sinon l'extraction des nouveaux termes pôles et leur stockage.
3. Le re-calcul de nouveaux termes pôles et le stockage de cliques d'ordre 3.
4. Le re-calcul de nouvelles cliques d'ordre 3 et le stockage de cliques d'ordre 4.

5. L'union de cliques d'ordre 4 et l'agrégation de termes.
6. Le stockage de nouveaux clusters sous la condition du maximum de termes en commun.

Le processus incrémental assure un gain en temps de calcul. Ainsi l'extraction des termes pôles est $O(N_0/3)$ en temps de calcul et la recherche de cliques d'ordre 3 est $O(N_0^3/20^3)$ en temps, N_0 étant le nombre de termes. La construction de cliques d'ordre 4 est $O(n(n-1)*11/2)$ en temps avec $n=N_{3t}*3/N_0$, N_{3t} étant le nombre de cliques d'ordre 3. Puisque le nombre de termes communs entre deux clusters est inférieur à 4 nous forçons les cliques d'ordre 4 à participer une fois dans la construction des clusters. Ainsi la construction des clusters est $O(n.\log(n))$ en temps avec $n= N_{4t}\ 3/N_0$, N_{4t} étant le nombre de cliques d'ordre 4. L'application qui peut supporter l'allocation et la désallocation de mémoire permet au traitement de n'être pas plus que polynômiale en temps de calcul $O(n^3/20^3)$. Avec cet ordre d'incrémentalité le traitement devient $n^*(n+n^*)^2/20^2$ où n^* est le nombre de nouveaux termes pôles.

5. EXPERIENCES ET DISCUSSION

5.1. PROTOTYPE

Le prototype fonctionne exclusivement en français mais peut être adapté à l'anglais. Le moteur de clustering a été écrit en langage C et l'interface utilisateur en Java. La Figure 5 montre l'interface utilisateur.

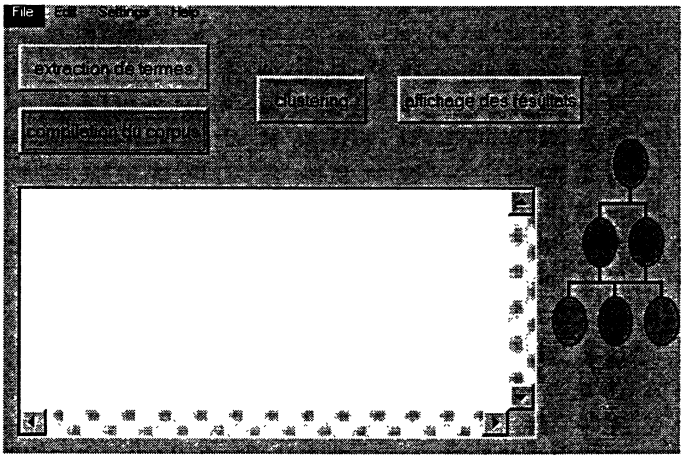


Figure 5. Panneau principal de l'interface utilisateur.

Les étapes sont activées de la gauche vers la droite avec 4 boutons. Le premier bouton permet à l'utilisateur de sélectionner un corpus et alors d'activer l'extracteur statistique. L'action du bouton en bas à gauche compile le corpus et le fichier de termes résultant de la première action pour aboutir à un fichier position, un fichier de verbes et un fichier de termes avec des termes simples discriminants. Le bouton central active le calcul de cluster. Finalement le bouton de droite ouvrira une fenêtre Netscape pour afficher la liste des termes pôles.

L'exécution du clustering a besoin de certains paramètres. Une fenêtre d'administration propose tous les paramètres que nous allons décrire.

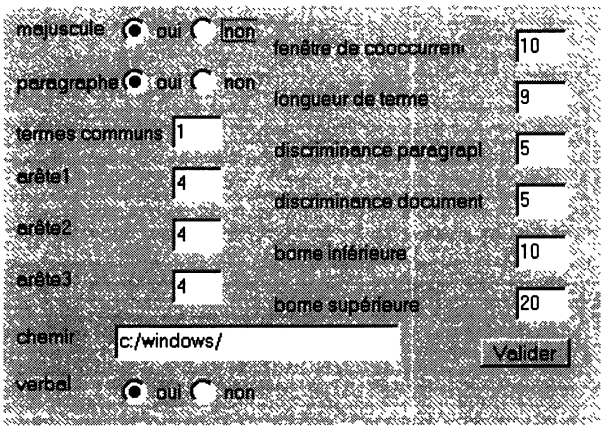


Figure 6. Panneau d'administration.

Le premier paramètre permet l'identification de noms propres dans la compilation de corpus. Le 2ème paramètre est l'identification de paragraphes pour confiner l'association de co-occurrences dans un paragraphe. Le 3ème paramètre est le nombre de termes communs autorisé entre deux clusters. Les trois paramètres suivants sont les seuils d'arité pour sélectionner les cliques d'ordre 3. On trouve ensuite le chemin pour stocker le fichier d'initialisation. Un mode verbal est proposé pour grouper ou non en tenant compte du fichier de verbe. En haut à gauche on peut ajuster la fenêtre de co-occurrence. Le 9ème paramètre est la largeur maximale du terme en nombre de mots pour l'identification de formes variantes. Le 10ème paramètre est un paramètre de discrimination pour conserver les mots les plus discriminants par paragraphe. Le 11ème paramètre est identique au précédent mais par document. Les deux derniers paramètres sont les fréquences bornes minimum et maximum de l'heuristique d'extraction des termes pôles.

5.2. TEST

Nous avons traité quatre corpus dont les caractéristiques sont résumées dans la table 6. Nous examinons trois corpus avec différents contenus (A, B, C) et un dont le contenu est mixte (D).

Les corpus sont formés en compilant des textes libres sauf le corpus B qui rassemble des bilans médicaux sur la coronarographie en cardiologie. Ce corpus est plus complet dans son domaine que les autres mais il ne possède pas de signes diacritiques. Il faut noter qu'aucune «désambiguïsation» n'est faite pour générer le fichier de verbes et la lemmatisation des groupes nominaux. Notre algorithme basé sur des heuristiques travaille avec un dictionnaire de 170 000 formes de mots français et un choix dichotomique des formes lemmatisées. De ce point de vue le corpus B est plus complexe à cause de l'absence des signes diacritiques accusant une source d'ambiguïté. Une étude montre que 80 % des mots ambigus sont confinés dans une liste de 100 mots. Par exemple «nourrissons» (du verbe nourrir) et «nourrissons» (bébés) ou «été» (être) et «été» (saison). Généralement les deux formes ambiguës n'apparaissent pas ensemble dans un corpus spécialisé dû à la fermeture du champ sémantique. Par exemple en aéronautique entre «avions» (verbe) et «avions» (appareil) dans l'expression «avions de combat» la lemmatisation conduit à «avoir de combat» au lieu de «avion de combat» mais dans le texte l'auxiliaire «avoir» n'interfère pas avec combat. Ainsi même si la lemmatisation n'est pas correcte cela correspond toujours à la même expression. La «désambiguïsation» est un moyen plus efficace pour lemmatiser les verbes, les rassembler correctement, et les sélectionner par fréquence du fait que l'on stocke les verbes ayant une fréquence

inférieure à 10 dans un fichier. Cette heuristique fait que l'algorithme converge en temps et en consommation mémoire. Mais en général 95 % des verbes ont une fréquence inférieure à 15 dans les corpus de la taille que nous avons traitée. Deuxièmement les verbes fréquents sont dangereux à considérer parce qu'ils sont plus souvent utilisés dans différents contextes, comme : avoir, être, pouvoir, mettre, devoir, faire...

Corpus	Domaines	Nombre de tokens	Nombre de tokens différents	Nombre de termes
A	Aéronautique	28000	3400	540
B	Rapports médicaux	30000	2900	510
C	Histoire russe	68000	6400	880
D	Aéronautique et histoire russe	51000	5800	630

Tableau 6. Données sur les corpus

Le tableau 7 montre quelques caractéristiques concernant les résultats. L'algorithme actuel n'est pas incrémental.

Corpus		Nombre d'instances	Instances multiples	Nombre de clusters	Nombre de termes pôles
A	Direct	11	0	2	2
	Mode verbal	340	78	56	44
B	Direct	804	165	146	89
	Mode verbal	626	140	101	75
C	Direct	183	31	35	33
	Mode verbal	681	158	122	98
D	Direct	37	6	6	6
	Mode verbal	386	84	70	61

Tableau 7. Données sur les clusters.

Le traitement a été exécuté sur un pentium avec 64 Mo de Ram. Le temps d'exécution est entre une demi-minute pour le plus petit corpus et 3 minutes pour le plus gros dans le mode verbal. Dans le mode direct, le temps est plus long de 2 à 6 minutes. Le traitement est lent essentiellement à cause du traitement des étapes qui ont lieu en même temps et sont stockées en Ram. Le temps d'exécution du clustering devrait être divisé par 10 avec une implémentation incrémentale.

Nous avons remarqué empiriquement que le nombre de clusters est entre $2\sqrt{N}$ et $5\sqrt{N}$ où N est le nombre de termes. Nous observons qu'un fichier composé seulement de multitermes conduit à une très petite quantité de clusters voire à aucun. Depuis que les clusters sont construits par associations circulaires, un terme peut être associé avec un autre via un troisième. Les mots simples sont plus fréquents que les multitermes ainsi ils génèrent des associations utiles pour lier des multitermes entre eux. Les instances multiples couvrent plus de 25 % de toutes les instances. Une instance multiple est une instance qui prend part à plusieurs clusters. Cela signifie qu'un terme révèle

significativement plusieurs contextes. Deux contextes différents ne sont pas nécessairement deux contextes de deux thèmes différents mais deux facettes d'un contexte. Par exemple «pays russe» est impliqué dans deux clusters thématiques différents, un sur les aspects politiques et le second sur les aspects géographiques :

1. Pays russe, état russe, permanence, domination mongole, et
2. Pays russe, essence, migration, toundra.

Mais «Mer Noire» est impliqué dans des clusters sur le rôle politique de la «Mer Noire» :

1- Périphérie, mer Noire, menchevik, Alexandre 1er, municipalité, quartier général, et

2- Tartares de Crimée, mer Noire, campagne électorale, fin de la seconde guerre mondiale, même moment, seconde guerre mondiale, côte de la mer Noire.

Maintenant nous allons analyser le graphe qui donne naissance à un cluster. Le cluster est tiré des résultats du corpus A.

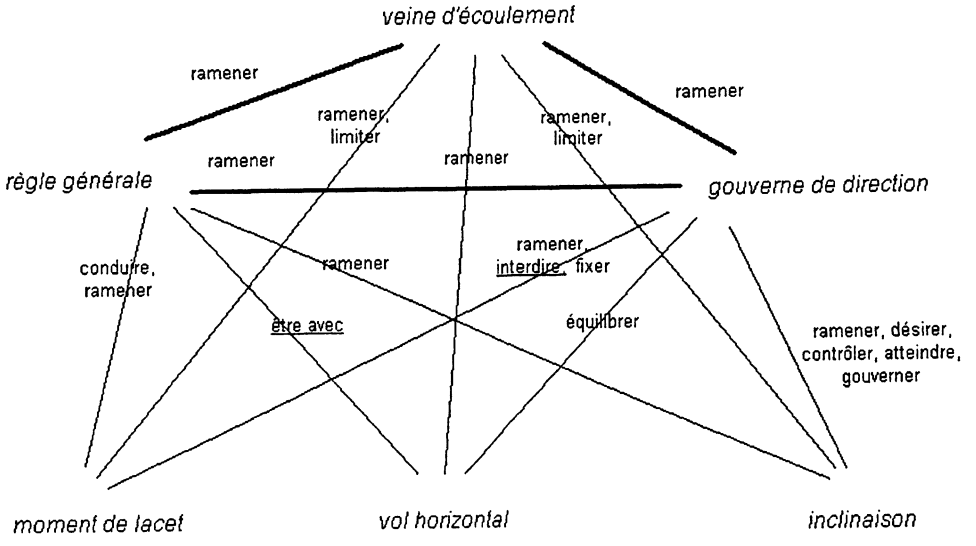


Figure 7. Représentation du graphe d'un cluster

Il a 6 instances réparties en 26 occurrences. Ci-dessous, les instances sont listées avec leurs formes variantes (VF) et leurs fréquences.

- veine d'écoulement (2)
- règle générale (4)
- gouverne de direction/ VF «gouvernes de profondeur et de direction» (7)
- moment de lacet / VF «moments de roulis, de tangage, et de lacet» ; «moment stabilisant de lacet» (4)
- vol horizontal/ VF «vol rectiligne horizontal» (5)
- inclinaison (4)

Nous listons les verbes contextuels pour chaque terme du cluster :

- veine d'écoulement : délimiter, établir, devenir, se composer de, monter, limiter, ramener, préparer, mettre, appeler, annuler, installer

- règle générale : être avec, s'élever, tendre, ramener, stabiliser, rendre, noter, servir, rapprocher, augmenter, compter

- gouverne de direction : créer, annuler, induire, lier, ramener, désirer, conserver, contrôler, atteindre, interdire, refuser, causer, gouverner, fixer, équilibrer, appliquer, présenter

- moment de lacet : limiter, agrandir, résulter, établir, fixer, générer, ramener, tendre, produire, permettre, interdire, accomplir, imposer, gouverner

- vol horizontal : équilibrer, être avec, apparaître, montrer, ramener, garder, arranger, arrêter, prendre place, excéder, dépendre, converger, perturber, pousser, ajuster

- inclinaison : pousser, contrôler, désirer, maintenir, ramener, atteindre, gouverner, limiter, dépasser

Nous avons remarqué qu'un très petit nombre de verbes représentent la terminologie du domaine d'un corpus. Par exemple, sur les 345 verbes du corpus B concernant les maladies du cœur seulement «revasculariser», «hospitaliser» et «ponter» sont typiques. Cela ne biaise pas du tout le développement de schéma dans les phrases et les paragraphes entre entités et verbes. Par exemple dans le même corpus, un schéma typique utilise le verbe «montrer» entre une analyse servant au diagnostic : «coronarographie», et une lésion, «coronarographie» + «montrer» + «lésion».

Une opération de généralisation se déroule après le calcul des clusters. Comme cela a été expliqué au paragraphe 4.2, la généralisation s'accomplit en deux étapes. Nous présentons des étiquettes sémantiques extraites du thésaurus d'après le triplet (terme pôle, terme pivot 1 et terme pivot 2). Nous devons préciser qu'un cluster ne peut pas recevoir d'étiquette sémantique si aucune correspondance avec les éléments du thésaurus n'existe.

- Généralisation à la 1ère étape :
  - Combustibilité : turbine à gaz, transport aérien international, transport international
  - Mélange : alliage, vitesse rapide, processus
  - Organisation : processus, travaux, disque
  - Participation : coopération, caoutchouc mobile, avion de combat
  - Défense : détection électromagnétique, radar, mm
  - Détection : antenne tournante, radar, système anti-aérien
- Généralisation à la 2ème étape : À cette étape nous cherchons aussi les catégories les plus fréquentes. Les catégories émergentes sont :
  - fréquence 4, (code 870) sports
  - fréquence 4, (code 820) transport par air
  - fréquence 4, (code 656) défense.

Ainsi, les thèmes dominants sont le sport, le transport par air et la défense, compte tenu de nos heuristiques de pondération sur les codes. Ainsi nous généralisons les catégories trouvées à l'étape 1 pour chacun des superclusters. La généralisation peut viser un ou plusieurs superclusters. Pour quelques superclusters pris comme instances, nous trouvons les catégories suivantes de 2ème niveau : *ordre et mesure* (2), *temps* (1),

impulsion (1), *matière* (3), *action* (2), *guerre et paix* (1), *activités commerciales* (3), où l'on a indiqué, entre parenthèses, le nombre de superclusters concernés.

5.3. ÉVALUATION

En général, l'évaluation est un problème difficile. Dans notre cas, l'évaluation est intéressante seulement quand on utilise des données du monde réel. Des données artificielles non triées sont mal adaptées pour observer une relation entre les résultats et les données. En outre, on peut générer un texte avec une distribution aléatoire de termes mais comment générer un texte ayant un sens avec des associations idéales de termes que l'on se donnerait par avance ? Ainsi la complexité de l'évaluation est plus élevée que l'estimation de l'efficacité avec des données artificielles désordonnées. Nous procédons de manière à utiliser une série de classes de termes prise comme référence d'un domaine thématique. La déviation par rapport à ce cadre sera effective en combinant 3 paramètres : p, r et α, résultant de $T = \alpha.p.\sqrt[3]{r}$ (Turenne & Rousselot, 1998). Ce paramètre T est très similaire à l'évaluation d'un système de recherche documentaire (information retrieval) avec les paramètres de rappel et de précision.

Dans notre expérience r représente le nombre de termes de la même catégorie (parmi 9 catégories) divisé par le nombre d'items de la taxinomie des sous-classes identifiées dans le cluster. L'autre paramètre p représente le nombre de termes de la même catégorie (parmi les 9 catégories) mais divisé par le nombre d'items du cluster. Les 9 catégories sont : physiologie cardiovasculaire, symptomatologie, anatomie coronarienne, pathologie générale, pathologie coronarienne, facteur de risque, diagnostic, thérapie, information patient. Le paramètre T reflète de bonnes corrélations quand p et r sont ensemble élevés ou dans le cas où p est petit et r est élevé ou vice versa. Pour éviter les bons clusters de petite taille nous choisissons de réduire T par un facteur de taille : $\alpha = 1 - \exp[-(2_{*}n/5)2]$, n étant le nombre d'items d'un cluster. Pour les plus petites tailles $n = 4$ et $\alpha = 0.92$, $\alpha = 1$ pour les autres tailles. La méthode d'évaluation que nous utilisons peut être qualifiée de semi-empirique puisque nous utilisons une classification manuelle liée au domaine du corpus. Ainsi le paramètre T doit être considéré comme un indice de large échelle par rapport aux données du monde réel et non comme un indice absolu d'évaluation. Nous avons synthétisé les résultats sur deux diagrammes concernant le nombre de clusters par intervalle de valeurs de T (Figure 8).

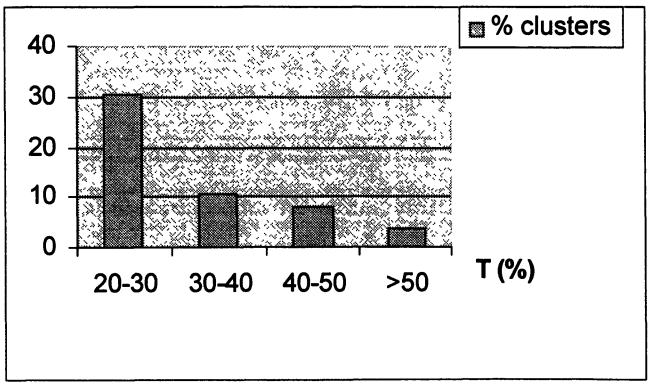


Figure 8. Nombre de clusters par intervalle de T

et le nombre de clusters par intervalle de valeur de P (Figure 9).

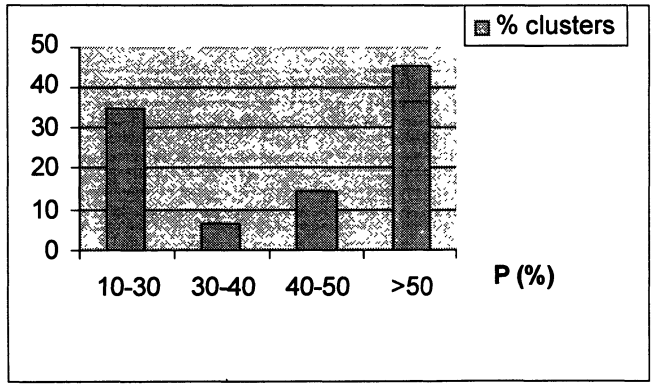


Figure 9. Nombre de clusters par intervalle de P

Premièrement nous observons qu'une majorité de clusters contient plus de 30 % de termes d'une même catégorie. Cette observation montre d'une part que les associations d'un cluster ne sont pas aléatoires. D'autre part, cela montre que le cluster se décompose en deux composantes : la 1ère composante réunit des termes d'un catégorie principale et la 2ème composante réunit des termes d'autres catégories liées au contexte de la 1ère composante.

Cela peut être expliqué par la construction du cluster qui regroupe deux termes pertinents de la même catégorie grâce à des termes typiques du contexte ; par exemple «lésion» et «anomalie» seront agrégés par «coronarographie» mais ensuite «coronarographie» sera gardé à l'intérieur du cluster. À l'heure actuelle, nous n'avons pas d'heuristique pour discriminer «coronarographie» et seulement garder «lésion» et «anomalie». Dans Turenne & Rousselot (1998), nous décrivons cette méthodologie d'évaluation pour comparer quatre méthodes de clustering non supervisées : les réseaux de neurones de Kohonen, le clustering hiérarchique agglomératif ascendant avec une distance euclidienne, le clustering hiérarchique agglomératif descendant avec une distance de khi2 et le clustering des mots associés. Les résultats de cette étude montrent la présence d'une très faible quantité de clusters satisfaisant la contrainte T > 40 % . Seulement 1 % des clusters (1 cluster) pouvaient valider cette contrainte, celui-ci étant lié à trois médicaments cités dans la même phrase plusieurs fois (comme une prescription médicale). Nos résultats semblent manifester un comportement équivalent de qualité décroissante quand T augmente. Mais la quantité de bons clusters augmente. D'après notre méthode, nous obtenons 12 % de clusters satisfaisant la contrainte T > 40 % .

Voici certains des meilleurs clusters :

- T = 0.63, thérapie/médicaments — Ergométrine, mg, voie intraveineuse, bolus.
- T = 0.52, anatomie cardiovasculaire — Artère circumflexe, branche, circumflexe, distal, bas, marginal, artère gauche, paroi gauche, ventricule gauche.
- T = 0.51, thérapie/médicaments — Injection, atropine, mg, nitroglycérine.
- T = 0.50, anatomie cardiovasculaire — Inflation, territoire, territoire. Coronarographie (artefact de l'extraction de termes), territoire latéral.
- T = 0.48, diagnostic — Cine-ventriculographie gauche, cathéter, électrocardiogramme, dérivation.

Nous observons une large présence d'instances liées à la pathologie dans 61 clusters sur 101 clusters (60 %). Par conséquent certains clusters peuvent être bien définis par rapport à leur thématique comme dans l'exemple mentionné ci-dessus, une vue générale du contenu de ces clusters est pathologie cardiovasculaire et plus particulièrement la pathologie analysable par coronarographie. Parmi les 15 techniques utilisées en cardiologie, la coronographie est une de celles dont on discute plus spécialement dans le corpus B.

Certains clusters de l'ensemble des 101 clusters sont rassemblés par un traitement des clusters. Ces clusters, en fonction de leur terme pôle, peuvent être fusionnés en superclusters de 1, 2 ou 3 clusters. Ainsi on compte 75 superclusters. Le terme pôle d'un supercluster a une certaine signification attachée à une certaine catégorie. Cela n'implique pas qu'un supercluster est lié au même thème mais se trouve décliné en une ou plusieurs catégories différentes ; ce phénomène caractérise une structure fine. Par exemple, le supercluster suivant traite d'anatomie coronarienne (AC) et de pathologie coronarienne (PC) et possède son terme pôle liée à l'anatomie. On dégage ainsi la structure fine AC ↔ AC/PC :

- Artère coronaire — Fréquence cardiaque, manifestation, myocarde, récent infarctus, infarctus du myocarde, maladies coronariennes, clinique, centre, degré.
- Artère coronaire — Artère gauche, fait, cathéter.
- Artère coronaire — Artère gauche, branche postérolatérale, coronaire gauche, artère droite.

Ensuite nous procédons à notre généralisation des superclusters pour obtenir une hiérarchie. De la même manière que nous avons décrit le traitement de la généralisation grâce à un thésaurus, nous collectons tous les codes de catégorie pour un supercluster donné et particulièrement des termes pôles et pivots. Nous déduisons le code du cluster pour un supercluster, celui correspondant au nom de catégorie de 1er niveau, et le nom de catégorie de 2ème niveau généralise la catégorie de 1er niveau (Tableau 8).

Le second processus de généralisation consiste à réunir tous les codes liés aux superclusters. Ensuite nous les classons par ordre décroissant pour attribuer les plus fréquents (3) à l'ensemble des superclusters (Tableau 9).

Un aperçu général de l'attribution des catégories montre les résultats suivants : 21 superclusters ont été catégorisés dans une catégorie médicale et 43 ont été classés dans une catégorie d'état, finalement 85 % ont été logiquement classés dans un thème relatif au contenu sémantique du corpus.

Le traitement du corpus D montre une discrimination des sujets développés dans le corpus. Le corpus a néanmoins deux composantes : l'aéronautique et l'histoire russe. Des 61 superclusters 27 se rapportent à l'histoire russe, 19 se rapportent à l'aéronautique et 15 sont ambigus. Un total de 46 superclusters – soit 75 % de la totalité – peuvent être attribués à leur thème respectif.

Termes des clusters	Code Pivot	Code pôle	Code cluster	Nom de catégories (niveau inférieur)	Nom de catégories (niveau supérieur)
Récent infarctus, ecg, Infarctus, coronarographie, infarctus inférieur	0	0	0	0	0
Age année, adresse, confiance, Mr Moplgir, confiance I, anomalie, urgence Age année, conclusion, patient année, tabagisme, jour, temps, thérapie anticoagulante, x	58/753	312*2/185/ 316/317/ 195	312	Age	Age de la vie
Marginal, aorte ascendante, cathéter, sténose gauche, sténose marginale, sténose gauche	391/331/622/ 623/679/681/	132/582*2/ 640	132	Côté	Frontière
Anomalie, spasme coronarien, thallium, pronostic, méthergin, femme, question, fonction gauche, segment, coronaire droit Anomalie, ventriculographie gauche, territoire, ecg, ventriculographie, coronaire droit	210/248/328/ 345/326/331/ 391/331	23/29/55	331	Cœur	Corps
Anticalcique, ischémie, ischémie du myocarde, cas, avk, spécialement	332/383/332/ 383/328/331	0	332	Sang	Corps

Tableau 8. Distribution des codes pour certains clusters

Nombre d'occurrences	Code de catégories	Nom de catégories (niveau inférieur)
31	383	Maladie
23	331	Cœur
21	391	Médecine
11	792	Travail
10	185	Période
10	392	Chirurgie
10	393	Soin du corps

Tableau 9. Catégories les plus fréquentes

6. CONCLUSION ET PERSPECTIVES

Imiter la capacité du raisonnement humain concernant la connaissance d'un domaine ou d'un utilisateur est difficile à modéliser de façon exhaustive. Des outils d'acquisition des connaissances pour le traitement de documents émergent massivement utilisant des techniques d'apprentissage (Faure & Nedellec, 1998 ; Fujihara, Simmons, Ellis & Shannon, 1997 ; Feldman & Dagan, 1997). Le clustering de termes est une voie possible pour se rapprocher de notre faculté humaine de catégorisation. Nous avons conçu un traitement entièrement automatique et séquentiel de données textuelles avec 3 opérations : une extraction de termes suivie d'un clustering et finalement d'une hiérarchisation. Cette séquence d'acquisition de connaissances permet d'extraire des unités de sens (termes) et de les structurer en unités de connaissance (clusters). Notre processus de construction de cluster est marqué par une monothétie, du fait qu'il est basé sur un échantillon de termes pôles et de couples de termes pivots. L'extraction de catégorie correspond assez bien à un apprentissage de concept. Nous définissons un concept à l'aide de son intension et de son extension. L'intension est fournie par l'interprétation qu'un utilisateur infère du contenu d'un cluster, des verbes contextuels et de la généralisation réalisée par le thésaurus. Le thésaurus crée des liens hyperonymiques avec les éléments d'un supercluster. L'extension est fournie par les éléments d'un supercluster ; un supercluster est l'union de certains clusters ayant même terme pôle. Depuis longtemps, les méthodes générales de clustering ont été utilisées sur le langage naturel. Le but de la méthode présentée dans cet article est de la rendre très proche de la structure des données d'entrée sur lesquelles elle s'applique. La bonne conduite du processus de clustering est sensible à trois phénomènes linguistiques sur l'interprétation des résultats et que nous avons étudiés : l'influence des expressions composées avec les mots simples, la réduction des formes variantes en formes canoniques et les relations de schéma entre parties du discours. Le regroupement de termes en subdivisions homogènes d'une hiérarchie peut être utilisé dans plusieurs domaines de gestion de l'information où l'extraction et la diffusion de documents doivent être rapide et efficace : le filtrage d'information (Hull & Pedersen, 1995), le routage de documents (Grobelnik & Mladenic, 1998) et la navigation dans les documents (Grefenstette, 1997 ; Hearst, 1994). Notre système appelé Galex peut être exploité comme interface à un système de représentation des connaissances pour remplir la base des connaissances et former un système ontologique. Nous poursuivons nos travaux dans l'implémentation de la généralisation qui est établie à la main pour le moment. Nous ne croyons pas qu'une structuration des connaissances puisse être réalisée par des processus strictement automatiques, nous avons donc prévu de développer une interface utilisateur de correction. Nous voulons aussi décrire la hiérarchie des catégories par un formalisme logique pour l'intégrer dans un système de représentation des connaissances.

REMERCIEMENTS — Nous remercions Dr Frey et Dr Barthel pour leur aide dans l'interprétation des clusters concernant les termes médicaux. Merci également à F. Rousselot, B. Migault, F. de Beuvron et I. Pevunov pour leurs commentaires. Nous remercions aussi la société filiale de Xérox, Inxight à Grenoble pour leur aimable prêt de l'extracteur de syntagmes nominaux. L'auteur est reconnaissant aux rapporteurs pour leurs suggestions et commentaires au cours de la préparation de cet article.

BIBLIOGRAPHIE

ASSADI, H., «Knowledge acquisition from texts: using an automatic clustering method based on noun-modifier relationship», *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics,* Madrid, 1997.

AUGUSTSON, J. G., MINKER, J., «Deriving term relations for a corpus by graph theoretical clusters», *Journal of the American Society for Information Science*, Vol. 21 n° 2, 1970.

AUSSENAC-GILLES, N., BOURIGAULT, D. and CONDAMINES, A., «How can knowledge acquisition benefit from terminology?», *Proceedings of 9th KAW,* Banff (Canada), 1995.

BASILI, R., PAZIENZA, T. and VELARDI, P., «Corpus processing for lexical acquisition», *Categorization of Lexical Units*, ed. B. Boguraev J. Pustejovsky, MIT Press, 1997.

BISSON, G., «Clustering and categorization», *Actes de CIMPA96,* Nice (France), 1996.

CAPPONI, N., TOUSSAINT, Y., «Interprétation de classes de termes par généralisation de structure prédicat-arguments», *Ingénierie des Connaissances* [*French Knowledge Engineering Workshop,* Pont-à-Mousson (France), 1998.

CARPINETO, C., ROMANO, G., «A lattice conceptual clustering system and it application to browsing retrieval», *Machine Learning*, n° 2495, 1996.

CHANOD, J.-P., TAPANAINEN, P., «Tagging French- comparing a statistical and a constraint-based method», *Proceedings of EACL'95*, Dublin, 1995.

CHANOD, J.-P., TAPANAINEN, P., «Creating a tagset, lexicon and guesser for a French tagger», *ACL-SIGDAT,* Dublin, 1995.

CUTTING, D., KARLGREN, J., «Recognizing text genres with simple metrics using discriminant analysis», *Proceedings of COLING'94*, Kyoto (Japan), 1994.

DEVIN, Ch., *Panlingua a universal subsurface language*, Technical report, Hawaï (USA), 1998.

EDMONDS, Ph., «Choosing the word most typical in context using a lexical co-occurrence network», *Proceedings 35th annual meeting ACL*, Madrid, 1997.

FAURE, D., NEDELLEC, C., «Asium: learning subcategorization frames and restrictions of selection», *Text mining workshop of ECML,* Chemnitz (Germany), 1998.

FELDMAN, R., DAGAN, I., «Knowledge discovery in textual databases (KDT)», *Proceedings of the 1st International Conference on Knowledge Discovery KDD-95*, Montréal, 1995.

FENG, C., COPECK, T., SZPAKOWICZ, S. and MATWIN, S., *Semantic clustering acquisition of partial ontologies from public domain lexical sources,* Technical Report, Ottawa, University of Ottawa, 1994.

FISHER, D., «Knowledge acquisition via incremental conceptual clustering», *Machine Learning*, 2, 1987.

FISHER, D., SCHLIMMER, J., *Models of incremental concept learning: a coupled research proposal,* Technical Report, Carnegie Mellon University (USA), 1997.

FREGE, G., *On sense and reference* [Trans. Max Black: *Translations from the philosophical writings of Gottlob Frege*], ed. Peter Geach and Max Black., Oxford, Basil Blackwell, 1892

FUJIHARA, H., SIMMONS, D., ELLIS, N. and SHANNON, R., «Knowledge conceptualization tool», *IEEE transactions on knowledge and data engineering*, Vol. 9, n° 2, 1997.

GREFENSTETTE, G., «SQLET: short query linguistic expansion techniques, palliating on-word queries by providing intermediate structure to text», *Recherche d'Information Assistée par Ordinateur RIAO,* Montréal, 1997.

GROBELNIK, M., MLADENIC, D., «Efficient text categorization», *ECML text mining workshop*, Chemnitz (Germany), 1998.

HABERT, B., NAULLEAU, E. and NAZARENKO, A., «Symbolic word classification for medium-size corpora», *Proceedings of Coling'96,* Copenhague, 1996.

HARRIS, Z., *Mathematical structure of language*, New-York, ed. Wiley, 1968.

HEARST, M., *Contextualizing retrieval of full-length documents. Technical report,* University of California, n° UCB/CSD 94/789, 1994.

IBEKWE-SANJUAN, F., *Processing for thematic trends mapping,* Technical Report, Grenoble, Université de Grenoble, 1996.

http 1 http://jedlik.phy.bme.hu/~gerjanos/HMM/node7.html

http 2 http://jedlik.phy.bme.hu/~gerjanos/HMM/node11.html

http 3 http://jedlik.phy.bme.hu/~gerjanos/HMM/node8.html

HULL, D., PEDERSEN, J., «Method combination for document filtering», *Proceedings of SIGIR'96*, Zurch (Switzerland), 1996.

KANTER, I., KESSLER, I., «Markov processes: linguistics and Zipf's law», *Physical Review Letters*, Volume 74, Issue 22, 1995, pp.4559-4562.

KOHONEN, T., *Self-organization and associative memory,* ed. Springer-Verlag, 1989.

KIRSTEN, T., «Relational distance-based clustering», *ILP'98 workshop,* Berlin, 1998.

LINGRAS, P., «Classifying highways:hierarchical grouping vs Kohonen neural networks», *Journal of Transportation Engineering*, Vol. 121, n° 4, 1994, pp. 364-368.

LEBART, L., SALEM, A. and BERRY, L., *Exploring textual data,* ed. Kluwer, Academic Publishers, 1998.

MAIKEVICH, N., «From information space to knowledge space; ontology on internet», *CAI'98 Russia* [Conference on Artificial Intelligence], Pushino (Russia), 1998.

MIKHEEV, A., and FINCH, S., «Towards a Workbench for Acquisition of Domain Knowledge from Natural Language», *Proceedings ACL student session*, Madrid, 1995.

MICHALSKI, R., «Knowledge acquisition through conceptual clustering. A theoretical framework and algorithm for partitioning data into conjunctive concepts analysis», *International journal of policy and informatics systems*, Vol. 4, n° 3, 1980, pp. 219-244.

MEILA, M., HECKERMAN, D., *An experimental comparison of several clustering and initialization methods*, Technical Report MSR-TR-98-06, Microsoft, 1998.

MEMMI, D., GABI, K. and MEUNIER, J.-G., «Dynamic knowledge extraction from texts by Art networks», *Fourth International Conference on Neural Networks and their Applications NeurAp'98,* Marseille, 1998.

MESSATFA, H., ZAIT, M., «A comparative study of clustering methods», *Future Generation Computer System*, n° 500, 1997.

OAKES, M., *Statistics for corpus linguistics,* ed. Edinburgh textbooks in empirical linguistics, 1998.

POLANCO, X., GRIVEL, L., ROYAUTÉ, J., «How to do things with terms in informetrics: terminological variation and stabilization as science watch indicators», *Fifth International Conference on scientometrics & informetrics*, Edited by M.E.D. Koening and A. Bookstein, Medford (NJ, USA), Learned Information Inc., 1995, pp. 435-444.

ROUSSELOT, F., FRATH, P., «Extracting concepts and relations from Corpora», *Proceedings of Workshop on Corpus-oriented Semantic Analysis European Conference on Artificial Intelligence ECAI'96*, Budapest, 1996.

SCHILLER, A., «Multilingual finite-state noun phrase extraction», *Proceedings of ECAI'96 conference*, Budapest, 1996.

SCHÜTZE, H., SILVERSTEIN, C., «A comparison of projections for efficient document clustering», *Proceedings of the Twentieth Annual ACM SIGIR Conference on Research and Development in Information Retrieval,* Philadelphia (USA), 1997.

SKUCE, D., MEYER, I., «Terminology and knowledge acquisition: exploring a symbiotic relationship», *Proceedings of 6th KAW*, Banff (Canada), 1991.

SMADJA, F., MCKEOWN, K., «Automatically extracting and representing collocations for language generation», *Proceedings of Conference ACL*, Pittsburgh (USA), 1990.

SPARCK-JONES, K., *Synonymy and Semantic Classification*, Edinburgh, ed. Edinburgh University Press, 1987.

TANGUY, L., THLIVITIS, T., «PASTEL : un protocole informatisé d'aide à l'interprétation des textes», *Actes du colloque Conférence Informatique et Langue Naturelle ILN'96,* Nantes (France), 1996.

TEIL, G., LATOUR, B., «The Hume machine: can association networks do more than formal rules?», *Stanford Humanities Review (SEHR)*, Vol. 4, Issue 2 : *Construction of the mind*, 1995.

THOMSON, K., LANGLEY, P., «Incremental concept formation with composite objects», *Machine Learning proceedings of the 6th international workshop*, Ed. Morgan Kaufmann, 1988, pp. 371-378.

TURENNE, N., ROUSSELOT, F., «Evaluation of four clustering methods in text-mining», *ECML workshop on textmining*, Chemnitz (Germany), 1998.

WÜSTER, E., *Die terminologische Sprachbehandlung. I: Studium Generale*, Jahrg. Heft, 4, 1991.

YAROWSKY, D., «Word-sense disambiguation using statistical models of Roget's categories trained on large corpora», *Proceedings of conference COLING'92*, Nantes (France), 1992.

ZYTKOW, J., ZEMBOWICZ, R., «Contingency tables as the foundations for concepts, concept hierarchies, and rules: the 49er system approach», *Fundamenta Informaticae*, n° 30, 1997, pp. 383-399.