Acquisition de vocabulaire Patient/Médecin

Yassine Motie

Académie de Montpellier
Université Montpellier II
Sciences et Techniques du Languedoc

MÉMOIRE DE

STAGE RECHERCHE DE MASTER M2 INFORMATIQUE

effectué au Laboratoire d'Informatique de Robotique
et de Micro-électronique de Montpellier

Spécialité : **AIGLE**

Acquisition de vocabulaire Patient/Médecin

par **Yassine MOTIE**

Date : **01 Septembre 2014**

Sous la direction de **Sandra BRINGAY, Jérôme AZÉ, Thomas OPITZ et Pascal PONCELET**

Remerciements

Je remercie mes encadrants Sandra Bringay, Jérôme Azé, Thomas Opitz et Pascal Poncelet pour leur patience, leur disponibilité ainsi que leurs précieux conseils qui ont guidé ma démarche dans ce travail de recherche.

Je remercie également les membres du jury et en particulier mes rapporteurs Rodolphe Giroudeau, Maguelonne Teisseire et Konstantin Todorov pour le temps accordé à la lecture du rapport et dont les remarques éclairées me seront utiles.

Table des matières

Introduction et motivations

Le foisonnement de l'information médicale dû à l'informatisation croissante des professionnels de santé à l'hôpital et hors de l'hôpital ainsi que le déploiement d'Internet, nous pousse à réfléchir sur une manière d'exploiter cette importante masse d'informations, notamment celle produite en parallèle des sites institutionnels comme celui de l'assurance maladie `http://www.ameli.fr/` ou de la Haute Autorité de Santé (HAS) `http://www.has-sante.fr/`. En effet, nous assistons à une explosion du « web de la santé ». Des centaines de sites proposent des informations médicales, plus ou moins objectives et bien référencées. Elles ne sont pas toujours authentifiées par des médecins qualifiés et pour certaines peu mises à jour. Même lorsque les sites et les articles sont de qualité, les outils communautaires intégrés dans ces sites (tweet, facebook, commentaires en ligne...) permettent aux internautes de partager des commentaires appropriés ou non. Dans le cadre de ce master, nous allons nous focaliser sur un type particulier de sites web médicaux, **les forums de santé**.

Ces forums de santé contiennent des informations hétérogènes qui posent un réel problème, lié à la difficulté de les indexer automatiquement ou semi-automatiquement. En effet, les méthodes de traitement automatique du langage naturel (TALN) et celles de fouille de textes (FT) qui sont généralement appliquées pour l'indexation reposent sur l'utilisation de ressources de type dictionnaires, thésaurus ou ontologies. Il en existe de très nombreuses dans le domaine de la santé, qui réunissent des ensembles de concepts médicaux, généralement sélectionnés par des experts du domaine. Ces ressources sont mises à jour régulièrement. On peut citer par exemple l'UMLS qui est un metathésaurus constitué d'un ensemble de concepts biomédicaux, ainsi que de leurs informations sémantiques. Ces informations sémantiques sont elles-mêmes hiérarchisées dans un réseau sémantique, qui permet d'organiser les types et les catégories sémantiques des concepts. De plus, l'UMLS (Unified Medical Language System) `http://www.nlm.nih.gov/research/umls/` contient un lexique spécialisé qui décrit les informations syntaxiques des termes utilisés Mc Cray [1993]. On peut citer également le MeSH (Medical Subject Headings Lipscomb [2000]) `http://www.ncbi.nlm.nih.gov/mesh/` édité par United.States.National Library of Medicine (NLM), dont le but premier était d'indexer les références bibliographiques biomédicales Lindberg and Schoolman [1986]. Ce thésaurus a été traduit en français par l'INSERM (Institut National de la Santé et de la Recherche Médicale `http://www.inserm.fr/`) et dispose de la plupart des concepts de l'UMLS.

Si les ressources précédentes sont efficaces pour indexer les textes écrits par les professionnels de la santé, elles montrent leurs limites dans le cas des forums de santé. Dans ces derniers, les messages sont généralement écrits par des patients ou leur proches. Ils sont peu rigoureux, avec des fautes d'orthographe, des abréviations, des mots d'argot, *etc.* Or, à notre connaissance, il n'existe pas de thesaurus disponibles contenant le vocabulaire de ces patients en français, ce qui nous incite à en construire un en se basant sur un corpus de textes. Par exemple, les patients utilisent souvent le terme *onco* lorsqu'ils parlent de leur oncologue. Comme *onco* n'existe pas dans les ressources médicales, si l'on cherche à indexer automatiquement tous les messages contenant une référence à l'*oncologue*, on oubliera ceux contenant *onco*. L'originalité de l'approche présentée dans ce travail de master est non seulement d'identifier les termes réellement utilisés par les patients mais également de les mettre en correspondance avec les termes utilisés par les professionnels de santé et qui sont déjà « codés » dans les dictionnaires, thésaurus ou ontologies médicaux.

Proposer une nouvelle méthode pour acquérir un vocabulaire patient/médecin représente un véritable défi. Notre approche est résumée par la figure 1. Nous avons utilisé comme corpus une collection de paires de question/réponse disponibles en ligne sur le site `http://masantenet.com/`. Ce site offre la possibilité à des patients de poser des questions, via leurs cellulaires ou un site Web et d'obtenir des réponses de médecins. Le langage utilisé par les patients est très familier contrairement à celui des médecins. Lorsque le médecin répond, il commence parfois par reformuler la question. On peut donc trouver *onco* dans la question du patient et *oncologue* dans la réponse du médecin. L'objectif consiste alors à rapprocher ces deux termes. Dans un premier temps, nous avons appliqué des méthodes statistiques, linguistiques ou mixtes pour extraire des *candidats termes* patient, qui après avoir été validés constitueront les entrées du thesaurus. Nous rapprocherons dans un second temps ces termes candidats de concepts médicaux connus. Par exemple, le terme *onco* correspondra au terme *oncologue* que l'on retrouve dans le MeSH.

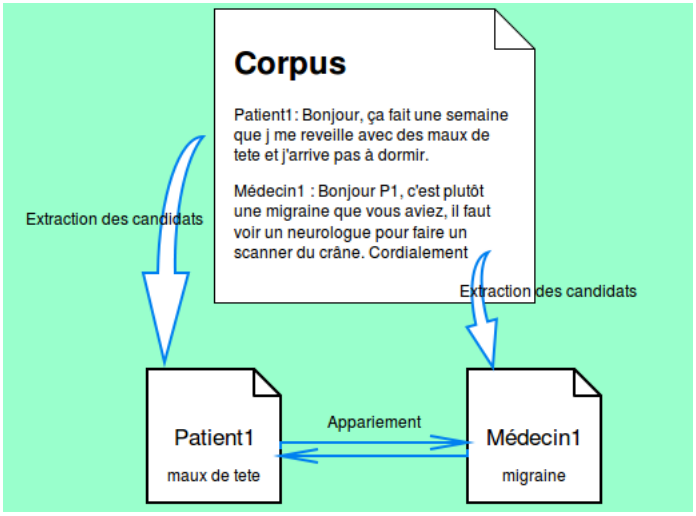

Figure 1 : Exemple du processus d'appariement de termes patient/médecin

Ce manuscrit sera organisé comme suit. Dans la section 1, nous présenterons un état de l'art en deux parties. La première concerne l'extraction de termes candidats et la deuxième concerne la mise en relation de ces termes candidats par des mesures de similarité. Dans la section 2, nous présenterons notre méthodologie. Dans la section 3, nous décrirons les expérimentations réalisées sur un jeu de données réelles et finalement nous conclurons et donnerons des perspectives dans la section 4.

Section 1 : État de l'art

Dans cet état de l'art, nous avons investigué différentes méthodes permettant d'extraire le vocabulaire des patients (candidats patient CP) et des professionnels de santé (candidats médecin CM). Ces derniers sont faciles à repérer et pour cela, nous utiliserons des ressources médicales existantes et les projetterons sur les textes de notre corpus. Pour les candidats CP la tâche est plus complexe. Un candidat patient CP peut être une chaîne de caractères (mot, syntagme, suite de caractères) qui constitue une entrée sélectionnée pour notre ressource car fréquemment utilisée par les patients mais qui est non répertoriée dans les thésaurus médicaux.

Dans une première partie de cet état de l'art, nous allons présenter les différents types de candidats possibles : mot, lemme, radical, n-grammes, *etc.* Dans une seconde partie, nous montrerons comment extraire les candidats CP, puis comment il est possible de les relier dans une troisième section.

1 Choix des candidats

Nous avons considéré comme candidats toutes les expressions fréquemment utilisées par les patients mais non répertoriées par les thésaurus médicaux. Pour capter ces expressions, nous nous intéresserons aux mots, aux lemmes et radicaux associés et aux n-grammes.

1.1 Mot, Lemme, Radical

On distingue deux types de mots : les mots invariables (*e.g* adverbes, interjections, conjonctions, prépositions) et les mot variables (*e.g* adjectifs, substantifs (ou noms), articles, pronoms, et verbes). Les mots variables ont la propriété de pouvoir être conjugués ou présentés sous différentes formes. On parle de formes fléchies du mot.

Généralement, dans les approches d'indexation, on ne considère pas tous les mots possibles, toutes les formes fléchies présentes dans les textes à indexer mais les lemmes. Le mot "abdominales" a comme lemme "abdominal" et le mot "digestive" a comme lemme "digestif". Ces lemmes sont souvent utilisés comme entrées de dictionnaire, ayant comme intérêt la réduction du nombre de candidats utilisés. On appelle lemmatisation l'opération qui permet de passer des mots aux lemmes.

Une flexion se décompose en deux entités : un radical et un (ou des) affixe(s). Comme la lemmatisation la racinisation vise à réduire le nombre de candidats et à rapprocher ceux ayant une base commune. La seule différence par rapport aux lemmes est que les racines ne sont pas forcément des mots de la langue. Par exemple le mot "oncologue" a pour radical (en anglais *stem* tel qu'employé dans [Lovins, 1968]) "onco" qui ne correspond pas à un mot réel. L'introduction du radical dans le domaine de la recherche d'informations a été faite par Porter [1980].

Ces types de candidats ne permettent pas de résoudre le cas des termes polysémiques, comme le mot "avocat" ayant un certain nombre de sens distincts selon s'il est employé avec "manger" ou "appeler". Un type de candidat peut permettre de lever en partie les ambiguïtés sémantiques : les n-grammes qui sont présentés ci-dessous.

1.2 n-grammes

Un n-gramme est une sous-séquence de n éléments construite à partir d'une séquence donnée. La notion de bi-grammes et trigrammes (c'est-à-dire avec respectivement n=2 et n=3) est apparue dans Pratt [1939] puis la notion de n-grammes de caractères dans Shannon [1948] pour la prédiction de caractères en fonction de ceux précédemment entrés. Le n-gramme de A sera égal à la séquence de n caractères consécutifs de A. A peut alors être un caractère ou bien un mot.

n-grammes de caractères Les n-grammes de caractères prennent en considération les espaces. Par exemple la représentation du mot "oncologie" en tri-grammes de caractères sera "onc","nco","col","olo","log","ogi","gie", pour le mot "mal de tête" la représentation en bi-grammes de caractères sera "ma","al","l ","d","de","e "," t","tê","êt","te". Ce type de candidats à plusieurs avantages parmi lesquels la non nécessité d'employer des descripteurs de type "radical". En effet, la description d'un corpus par les n-grammes de caractères prend automatiquement en compte les racines des mots les plus fréquents.

n-grammes de mots La première publication sur ce type de candidats date de 1979 Solso et al. [1979]. Les n-grammes de mots sont utilisés pour désambigüiser des mots composés. Par exemple "mal de tête" sera considéré comme un seul candidat avec des tri-grammes de mots. donc il faut dire que tout ces types de candidats sont plus ou moins pertinents car leur jointure nécessite une certaine rigueur De cette définition des n-grammes de mots découle différentes catégories de candidats dont **les syntagmes**, appelés aussi collocations, définis dans Clas [1994] comme étant un groupe de mots ayant un sens global déductible des unités (mots) composant le groupe. Par exemple, "fissure anale" est considéré comme une collocation car le sens global de ce groupe de mots peut être déduit des deux mots "fissure" et "anale". Ils se distinguent ainsi des *combinaisons figées* dont le sens ne peut pas être déduit de chacun des mots comme par exemple l'expression "tirer son chapeau" et des *combinaisons libres* comme "mal de tête" dont le sens des mots les composant n'est pas limité. En effet, on peut avoir mal à plusieurs organes.

Ces syntagmes peuvent être dans certains cas des entités nommées, comme les noms des personnes, de maladies, de médicaments et d'organisations, des expansions d'acronymes dont l'ordre dans lequel sont présentés les composants de cette expansion est important.

1.3 Synthèse

Un exemple de résultat de génération de candidats sur un document ne contenant que la phrase "Les accidents vasculaires cérébraux" :

- Par **mots** : "Les", "accidents", "vasculaires", "cérébraux".
- Par **lemmes** : "le", "accident", "vasculaire", "cérébral".
- Par **racines** : "l", "accident", "vascul", "cérébr".
- Par **concepts** : "A.V.C.".
- Par **bigrammes de caractères** : "l", "le", "es", "s ", " a", "ac", "cc", "ci", "id", "de", "en", "nt", "ts", "s ", " v", "va", "as"... "au", "ux", "x ".
- Par **bigrammes de mots** : "les accidents", "accidents vasculaires", "vasculaires cérébraux".
- Par **bigrammes de lemmes** : "le accident", "accident vasculaire", "vasculaire cérébral".
- Par **mot/catégorie grammaticale** : "les/DET :ART", "accidents/NOM", "vasculaires/ADJ", "cérébraux/ADJ".

Comme mentionné auparavant, on s'intéressera par la suite aux messages écrits dans des forums de santé, peu rigoureux, et écrits dans certains cas avec le language SMS. Dans la section suivante, nous nous intéresserons aux différentes manières d'extraire les candidats précédemment cités comme étant pertinents dans le cadre de notre étude.

2 Extraction des candidats

Il existe de très nombreuses approches d'extraction de termes candidats. Dans la suite, nous ne présentons que celles qui sont pertinentes pour notre étude.

Tout d'abord, nous considérerons les informations endogènes issues des candidats eux-mêmes pour extraire un premier type de candidat. Nous considèrerons ensuite les mots inconnus, non présents dans les lexiques pour définir un deuxième type de candidat. Pour finir, nous considérerons des mesures statistiques.

2.1 Candidats endogènes

Le langage médical a la faculté d'agglomérer des préfixes et des suffixes, créant ainsi un grand nombre de mots dérivés. Par exemple, les suffixes sont souvent des indicateurs d'états pathologiques. Les mots ayant le suffixe "ite" désignent souvent une inflammation comme *la pancréatite, appendicite*. Un traitement particulier utilisé dans Grabar and Zweigenbaum [1999] permet de segmenter un mot en composants pour voir si l'un des composants est un préfixe ou un suffixe identifié, se basant sur un ensemble de suffixes/préfixes fréquents des mots médicaux. Ainsi une acquisition des paraphrases pour des termes médicaux pourra se baser sur le système DériF Namer [2003] qui est un analyseur du lexique morphologiquement construit du Français et qui analyse non seulement les unités du lexique construites par dérivation (c'est-à-dire suffixées : SCOLAIRE/ADJ, préfixées APPAUVRIR/VERBE) mais aussi celles formées par composition savante ou néoclassique, par exemple :

- **myocardique** : myo=muscle, carde= cœur.
- **cholécystectomie** : cholécysto=vésicule biliaire, ectomie=ablation.
- **acromégalie** : acr=extrémité, mégal=grandeur.

Les mots à analyser pourront donc être issus du corpus des médecins "CM". Ainsi, le résultat de cette analyse pourra être projeté sur le corpus des patients "CP", notamment pour repérer des mots contenant des préfixes ou suffixes médicaux comme par exemple pour le mot "gynéco" utilisé par les patients pour représenter "gynécologue".

2.2 Mots inconnus

Afin de repérer les fautes d'orthographes fréquentes commises par les patients, on ciblera les mots inconnus du lexique, on regardera si une flexion possible de ce mot est présente (par substitution d'expression régulières, si le mot "Synapses" est absent, on essaiera de trouver le mot "Synapse"). Sinon, on regarde si une segmentation du mot en deux composants est possible (par exemple "ACTINOMYCOSE" pourrait être décomposé en "Actino" et "Mycose"). Sinon, nous rechercherons les mots s'orthographiant de la même manière à un caractère près. On s'appuyant sur le calcul du nombre de caractères qui diffèrent entre deux chaînes (distance de Hamming) ou la recherche de la

plus grande sous-séquence commune Navarro and Baeza-Yates [1999] ou la distance d'édition (notée E) Levenshtein [1966]. Cette dernière correspond à la somme minimale du coût des opérations à effectuer pour transformer une chaîne de caractères en une autre. Les opérations prises en compte sont la suppression, l'insertion et le remplacement de caractères.

La détection d'un mot inconnu produira la génération d'un message indiquant si une flexion du mot a été trouvée, s'il y a eu segmentation de mot, si un mot "voisin" a été trouvé ou si aucune correction n'a pu être mise en œuvre. Cette approche permet de repérer des :

— **erreurs orthographiques** : cyrhose –> cirrhose.
— **flexions** : anorexie mentale –> anorexie mental.
— **variations extra-morphologiques** : alpha lipoprotein –> alpha1 lipoprotein.

Ces mots sont la plupart du temps repérés dans le "CP" et pourront être apparié avec les mots bien écrits présents dans des connaissances exogènes (par exemple : le MeSH), en prenant en considération le contexte dans lequel ils apparaissent.

2.3 Sigles/définitions

La plupart des approches d'extraction des sigles et leurs définitions dans les textes s'appuient sur l'utilisation de marqueurs spécifiques (parenthèses, crochets, *etc.*). Un sigle est l'abréviation d'un groupe de mots formé, en général, par les initiales de ces mots. Une distinction existe entre les sigles dont chaque lettre est épelée (par exemple, EGC pour électrocardiogramme) contrairement aux acronymes qui sont prononcés comme des mots classiques (par exemple, BAT pour biopsie d'artère temporale).

La méthode de Roche and Prince [2007] consiste à identifier des acronymes et choisir la meilleure expansion de chaque acronyme dans un document qui ne contient aucune définition de celui-ci. Le corpus d'évaluation est constitué à l'aide de requêtes effectuées sur le moteur de recherche Google et en utilisant le moteur de recherche Exalead pour le calcul de la mesure. L'extraction des candidats sigles/definition pertinents s'appuie sur des marqueurs (parenthèses, crochets) et nécessite la prise en compte de deux traitements différents :

— *premier cas* : le sigle se situe avant la définition qui se trouve entre les marqueurs (les parenthèses dans le cas le plus courant). Exemple : "... R.G.O. (Réseaux gastro-oesophagien)..."

— *deuxième cas* : la définition se trouve avant le sigle qui se situe entre les marqueurs. Exemple : "...Réseaux gastro-oesophagien (RGO) ...". Dans ce cas, la taille de la définition est pour le moment indéterminable. Elle sera fixée à trois fois le nombre de lettres composants le sigle.

Cette phase retourne une quantité importante de bruit puisqu'elle s'appuie seulement sur les marqueurs tels que les parenthèses pour identifier un candidat potentiel. Ainsi, la nécessité d'effectuer un filtrage des candidats : alignement des lettres contenues dans l'acronyme avec les mots de la définition (si le premier caractère des mots de la définition ne peut être aligné les caractères qui suivent au sein des mots sont considérés). Les auteurs prennent l'exemple du sigle **JO** qui est associé aux deux définitions **Journal Officiel** et **Jeux Olympiques**. Ils construisent, dans un premier temps, manuellement un corpus constitué de l'ensemble de 100 pages web contenant le sigle. Dans un second temps Ils se basent sur un corpus se constituant de 1303 articles sur les lois de l'union européen où le sigle **JO** est cité. L'évaluation du système d'alignement se fait en s'appuyant sur les

données issues du site `http://sigles.net/` proposant 25463 sigles et leurs définitions issus de 17 langues.

La dernière phase consiste à utiliser des heuristiques spécifiques pour retenir les candidats pertinents. Ces heuristiques s'appuient sur le fait que les sigles ont une taille plus petite que leur définition, qu'ils sont en majuscule, que les définitions des sigles ont une longueur importante et ont tendance à posséder davantage de mots outils (par exemple, les articles et les prépositions), *etc.*

Cette méthode s'appuie sur des mesures statistiques (Information Mutuelle, Information Mutuelle au Cube, Mesure de Dice, détaillées dans la section mesure de similarité) et sur les résultats fournis par des moteurs de recherche, tout en prennant en considération le contexte de la page où le sigle a été trouvé. D'autres travaux se fondant sur la présence de marqueurs linguistiques souvent associées à des heuristique, comme dans Okazaki and Ananiadou [2006] qui utilisent des mesures statistiques (fondées sur l'approche C-value adaptée aux données biomédicales) pour l'extraction de la terminologie issue de domaine de spécialité (biomédecine). Ils s'appuient donc sur la fréquence des termes présents dans des corpus, contrairement à l'approche de [Roche and Prince, 2007] qui utilise le résultat de moteurs de recherche.

Les sigles apparaissent dans le "CM" accompagnés le plus souvent de leurs définitions. Dans le "CP", il arrive que les patients utilisent les sigles sans leurs définitions et sans prendre en compte leurs casses.

2.4 Approches statistiques

La sélection statistique de candidats est le type d'approche le plus répandu. Elle consiste en l'emploi de mesures statistiques afin de donner un score de qualité à un candidat, ce qui implique une sorte de classement par intérêt décroissant de ces candidats. Ainsi, seul les n premiers candidats sont généralement retenus après expérimentations.

Une première approche se fonde sur la loi de Zipf (Zipf [1941]) qui a montré que les termes les plus informatifs d'un corpus ne sont pas ceux apparaissant le plus dans ce corpus. Ces mots sont la plupart du temps des mots outils. Par ailleurs, les mots les moins fréquents du corpus ne sont également pas les plus porteurs d'informations. Ces derniers peuvent en effet être des fautes d'orthographe ou encore des termes trop spécifiques à quelques documents du corpus étudié. L'approche dite fréquentielle, notée *tf*, revient à considérer le nombre d'occurrences d'un terme i dans un document j (dans notre cas un document représentera un message de patient ou de médecin), ce qui nous mènera par exemple à considérer comme pertinents uniquement les candidats ayant un minimum et/ou un maximum d'occurrences. En revanche, cette méthode ne permet pas de distinguer un mot fréquent dans quelques documents d'un mot fréquent dans tout le corpus.

Une autre approche vient pondérer la méthode fréquentielle *tf* par le nombre de documents dans lesquels ce terme apparaît *df*. Le *tf-idf*, term frequency-inverse document frequency Salton and Yang [1973], peut se décrire comme suit pour un candidat i dans un document j (sera considéré comme document un message de patient) parmi les N documents du corpus.

$$\mathrm{W(ij)} : tf(ij) \times idf(i) \qquad (1.1)$$

avec

$$\text{idf(i)} : log\frac{N}{n(i)} \tag{1.2}$$

où n(i) est le nombre de documents dans lesquels apparaît le candidat i.

Une autre pondération possible est *Okapi* permettant d'obtenir de très bons résultats sur de nombreuses tâches de recherche d'information (RI). Elle a été proposée comme modèle de similarité dans un cadre probabiliste Robertson and Gatignon [1998], considérée comme un *tf-idf* prenant mieux en compte la longueur des documents. Sa définition est donnée dans l'équation (1.3) qui indique le poids du terme t dans le document d (k et b sont des constantes, dl la longueur du document, $dlavg$ la longueur moyenne des documents).

$$\mathbf{W}okapi(\text{t,d}) : tf(t,d) \times idf(t) : \frac{tf(t,d)*(k+1)}{tf(t,d)+k*(1-b+b*dl(d)/dlavg)} \times \log\frac{N-df(t)+0.5}{df(t)+0.5} \tag{1.3}$$

Ces deux mesures, *tf-idf* et *Okapi* ont été alliées avec la mesure C-value, dans Ventura et al. [2013]. L'approche C-value combine des informations statistiques (mesure c-value) et linguistiques (expressions régulières constituant des patrons selon la structure syntaxique des termes biomédicaux présents dans MeSH et UMLS). L'accent est mis sur la partie statistique. La partie linguistique se compose de l'étiquetage grammatical du corpus et d'un filtrage linguistique des candidats extraits. La mesure statistique C-value calcule la force de l'association d'un terme à des concepts du domaine, le but étant d'améliorer l'extraction des termes imbriqués. Cette mesure a été conçue spécialement pour l'extraction des termes multi-mots.

C-value(a) :

$$\begin{cases} w(a) * f(a) & if a \notin nested \\ w(a) * \left(f(t) - \frac{1}{|S(t)|} * \sum_{\tilde{t} \in S(t)} f(\tilde{t}) \right) & otherwise \end{cases} \tag{1.4}$$

avec *a* le terme candidat, *w(a) = log2(|a|)*, *|a|* le nombre de mots dans *a*, *f(a)* la fréquence de *a* dans un document, *Sa* la liste des termes contenant *a* et *|Sa|* le nombre de termes dans *Sa*. Sachant que les mesures *tf-idf* et *Okapi* associent à chaque terme du document un poids représentant sa pertinence dans le document par rapport au corpus auquel il appartient. Ils sont calculés avec un nombre variable de données, ensuite les poids obtenus sont normalisés et les termes ainsi regroupés dans un corpus unique afin de comparer les résultats. Puisque la précision dépendra de la méthode utilisée pour effectuer ce regroupement, trois fonctions ont été fusionnées pour calculer respectivement la somme (S), le max (M) et la moyenne (A) des valeurs des mesures du terme dans tout le corpus. D'où la génération de trois listes de *Okapi* et trois listes de *tf-idf*. De nouvelles combinaisons ont été conçues visant l'amélioration de la précision des termes extraits.

$$F\text{-}OCapi : 2 * \frac{OkapiX(a)*C-value(a)}{OkapiX(a)+C-value(a)} \tag{1.5}$$

$$F\text{-}tf\text{-}idf\text{-}C : 2 * \frac{TFIDFX(a)*C-value(a)}{TFIDFX(a)+C-value(a)} \tag{1.6}$$

a étant le terme, X le facteur $\in$ [S,M,A]. Par exemple, OkapiM(a) est la valeur obtenue pour la valeur maximale d'Okapi pour le terme a dans tout le corpus. La fréquence des termes dans l'équation (1.4) pourra être remplacée par une valeur plus significative (valeurs Okapi et Tfidf des termes).

C-Mx(a) :

$$\begin{cases} w(a) * Mx(a) & if a \notin nested \\ w(a) * (Mx(a) - \frac{1}{|Sa|} * \sum Mx(b)) & otherwise \end{cases} \quad (1.7)$$

Avec Mx(a) = [OkapiX|TfidfX], et X $\in$ [S,M,A].

Les approches statistiques pourront ainsi être appliquées afin de nous donner une idée sur les candidats les plus pertinents et le corpus où ils apparaissent le plus fréquemment. Ainsi, on applique cette approche sur le sigle *AVC* et sa définition "Accident Vasculaire cérébral". Le Tableau 1 montre la valeur *tf-idf* du sigle et de sa définition dans les deux corpus patient et médecin :

Candidats	Corpus Patient	Corpus Médecin
AVC	0.6	0.4
Accident Vasculaire cérébral	0.4	0.6

Tableau 1 : valeur *tf-idf* du candidat AVC/Accident Vasculaire cérébral

2.5 Approches syntaxiques vs Approches statistiques

Différentes approches d'extraction de la terminologie sont fondées sur des approches syntaxiques et/ou statistiques. Le système Termino de DAVID and Plante [1990] est un outil précurseur qui s'appuie sur une analyse syntaxique afin d'extraire les termes nominaux. Cet outil effectue une analyse morphologique à base de règles, suivie de l'analyse des syntagmes à l'aide d'une grammaire. Lexter de Bourigault [1994] et Syntex de Bourigault and Fabre [2000] s'appuient essentiellement sur une analyse syntaxique afin d'extraire la terminologie du domaine. La méthode consiste à extraire les syntagmes nominaux maximaux. Ces syntagmes sont alors décomposés en termes de "têtes" et d'"expansions" à l'aide de règles grammaticales. Les termes sont alors proposés sous forme de réseau organisé en fonction de critères syntaxiques.

La grande majorité des systèmes d'extraction de la terminologie est finalement mixte. Ainsi, Acabit de Daille et al. [1994] effectue une analyse linguistique afin de transformer les termes nominaux en termes binaires. Ces derniers sont ensuite triés selon des mesures statistiques. Le système Exit Roche [2004] permet quant à lui de sélectionner des termes binaires et/ou ternaires sur la base de critères linguistiques et/ou statistiques puis de construire itérativement des termes complexes.

Tonelli et al. [2012] combine quant à lui des mesures statistiques avec des annotations linguistiques en utilisant un analyseur morphologique afin d'extraire automatiquement les "multi-mots" les plus pertinents dans un corpus. Le corpus étant constitué de documents, le système extrait dans une première étape de chaque document les n-grammes de mots existants (en fixant la longueur des n-grammes sélectionnés). Dans une deuxième étape il extrait les "multi-mots" en appliquant deux mesures sur les n-grammes générés : *MinDoc* étant égale au nombre minimum des occurrences d'un n-gramme dans le document courant et *MinCorpus* représentera la même valeur par rapport

au corpus. Le n-gramme sera appelé "muti-mots" s'il apparaît *MinDoc* ou *MinCorpus* fois dans le document ou dans le corpus. Dans une troisième étape la force de chaque multi-mots (MW) est calculée en multipliant sa fréquence dans le document par sa fréquence dans le corpus. Les multi-mots sont donc triés, des patrons sont utilisés afin de ne conserver que ceux ayant des formes précises par exemple le patron [Nom][Préposition][Nom] a été utilisé. Pour éviter que les multi-mots les moins longs aient les fréquences les plus élevées ils seront classés selon certains paramètres : prenons par exemple accident, vasculaire et cérébral trois mots différents, si la fréquence du terme T1 = "accident vasculaire" est F(T1)=4 et celle de T2 = "accident vasculaire cérébral" est F(T2)=6 on aura :

— **subsomption des termes courts** : F(T2) = 10 et F(T1)=0 ce qui vise à supprimer les termes imbriqués.
— **renforcement des termes longs** : si F(T1)=6 et F(T2)=4 alors on calcule la moyenne qui sera égale à $\frac{F(T1)+F(T2)}{2}$ =5. On défini la pertinence comme suit : pertinence (T1) = F(AB)-moyenne=1 et pertinence(T2) = moyenne=5. On appliquant la *subsomption des termes courts* on obtient F(T1)=6 et F(T2)=6

Ces paramètres seront mis dans un fichier de configuration pour les activer ou les désactiver selon le besoin. Cette approche étant adaptable, basée sur un ensemble de paramètres ajustables aux types des documents et à la longueur préférée des termes, nous sera utile car les termes médicaux ont tendance à être long.

2.6 Étapes d'évaluation

L'étape d'évaluation des résultats suit souvent des critères quantitatifs et qualitatifs qui explicitent simultanément le nombre de candidats retournés et la pertinence des réponses.
Ceci revient à répondre à deux questions : A t'on oublié des candidats ? A t'on trouvé trop ?

On distingue le cas où les candidats retournés correspondent réellement à des termes médicaux (pertinents), et l'autre cas disjoint où ils n'y correspondent pas (non pertinents), comme le montre la figure 2 :

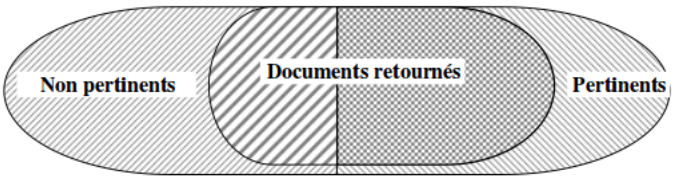

Figure 2 : Candidats pertinents, non pertinents, retournés

On cite alors les deux mesures : bruit (les candidats extraits non pertinents) et silence (les candidats pertinents non extraits) représentées dans la figure 3 comme suit :

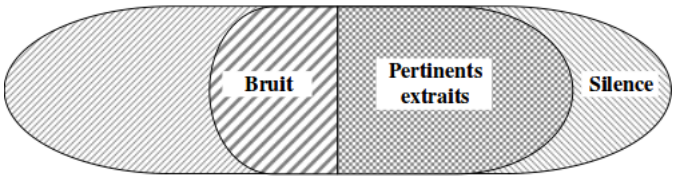

Figure 3 : Les mesures : bruit et silence

Deux mesures en découlent complémentaires des deux précédentes :

La précision qui représente le nombre de candidats pertinents extraits par rapport au nombre de candidats extraits

$$\textbf{précision} : \frac{\text{nombre de candidats retournés et pertinents}}{\text{nombre de candidats extraits}} \tag{1.8}$$

Le rappel, qui représente le nombre de candidats pertinents extraits par rapport au nombre de candidats pertinents

$$\textbf{rappel} : \frac{\text{nombre de candidats retournés et pertinents}}{\text{nombre de candidats pertinents}} \tag{1.9}$$

2.7 Synthèse

Une combinaison des approches citées auparavant pourra être effectuée dans le but d'extraire les candidats les plus pertinents du corpus "de patients" qui est assez bruité. On pourra aussi se baser sur les mots des médecins, grâce à un alignement des deux corpus de façon quasi parfaite, c'est-à-dire qu'il soit possible de mettre en regard le texte des médecins et des patients au niveau de la phrase, voire du syntagme ou du mot. L'idée est de voir jusqu'où peut aller une analyse purement automatique fondée sur une approche statistique s'appuyant sur la notion de n-grammes de mots (des séquences de n mots) ou de n-grammes de caractères.

Le tableau ci-dessous montre les différents types de candidats par rapport au corpus dans lequel ils apparaissent :

Types de Candidats	CP	CM	Approches
Mots médicaux	Oui (Céphalée)	Oui (Migraine)	Exogènes [MeSH, 1986]
Abréviations	Oui (onco)	Oui (oncologue)	Racinisation [Porter, 1980]
Mots d'argot	Oui (crabe)	Non (cancer)	Statistiques + Exogènes
Flexions	Oui (gene)	Oui (genic)	Etiqueteur morpho-syntaxique
Sigles/définitions	Non (RGO)	Oui (Reflux gastro-oesophagien)	Statistiques + Exogènes
Mots composés	Oui (mal de tête)	Oui (Encéphalopathie hépatique)	Patrons syntaxiques
Erreurs orthographiques	Oui (cyrhose)	Non (cirrhose)	[Levenshtein, 1966]

Tableau 2 : Tableau Candidats / Corpus / Approches

Après l'extraction des candidats pertinents des deux corpus, celui des patients et celui des médecins, on décrira dans la section qui suit les différentes méthodes susceptibles d'être utilisées afin d'apparier ces candidats construisant ainsi un dictionnaire liant les termes utilisés par les professionnels et ceux utilisés par les patients.

3 Mesure de similarité

Évaluer la similarité entre documents textuels est une des problématiques importantes de plusieurs disciplines comme l'analyse de données textuelles, la recherche d'information ou l'extraction de connaissances à partir de données textuelles (Text Mining). Dans chacun de ces domaines, les similarités sont utilisées pour différents traitements :

- en analyse de données textuelles, les similarités sont utilisées pour la description et l'exploration de données
- en recherche d'information, l'évaluation des similarités entre documents et requêtes est utilisée pour identifer les documents pertinents par rapport à des besoins d'information exprimés par les utilisateurs
- en Text Mining, les similarités sont utilisées pour produire des représentations synthétiques de vastes collections de documents

On distigue la similarité **syntaxique** et celle **sémantique**, la première étant une mesure permettant de comparer des documents textuels, en comparant des chaînes de caractères. C'est une métrique qui mesure la similarité ou la dissimilarité entre deux chaînes de caractères. Par exemple, les chaînes de caractères "Sam" et "Samuel" peuvent être considérées comme similaires. Une telle mesure sur les chaînes de caractères fournit une valeur obtenue algorithmiquement.

La similarité sémantique est un concept selon lequel un ensemble de documents ou de termes se voient attribuer une métrique basée sur la ressemblance de leur signifcation / contenu sémantique. Concrètement, cela peut être réalisé en définissant une similitude topologique, par exemple, en utilisant des ontologies pour définir une distance entre les mots, ou en définissant une similitude statistique, par exemple en utilisant un modèle d'espace vectoriel pour corréler les termes et les contextes à partir d'un corpus de texte approprié (co-occurrence).

Différentes variantes peuvent être traitées selon le type des candidats, que ça soit des variantes de caractères (casse, orthographe, accent) dans les mots, ou des variantes de l'ordre des mots, de mots vides, de mots morphologiquement proches mais formellement différents et de modifications morphosyntaxiques dans les termes. Les variations au niveau des caractères incluent :

Les variantes de casse sont faciles à traiter et à apparier. La mise en minuscules ou en majuscules des caractères n'est pas compliquée comme par exemple l'appariement du mot *migraine* côté patient et *Migraine* côté médecin. La seule confusion sera celle liant les noms propres et les noms communs, par exemple : *Pierre* et *pierre*.

Les variantes d'accentuation sont aussi aisées à traiter si l'on cherche à supprimer les accents. Par contre, si le but est leur rectification ou restauration, la tâche devient vite difficile. D'une part le repérage des règles d'apparition des caractères accentués Zweigenbaum and Grabar [2002] et d'autres part de désambiguïser en contexte les mots qui présentent différentes accentuations possibles Simard [1998]. On peut citer par exemple : l'omission d'accents : *céphalée - cephalée* ou les accents erronés : *céphalée - cèphalée*.

Les variantes orthographiques peuvent être enregistrées a priori, puis traitées et appariées en calculant par exemple le **String Matching**, qui est une mesure proposée par Maedche and Staab [2002] et qui repose sur la distance des chaînes (notée E) Levenshtein [1966]. Cette distance

correspond à la somme minimale du coût des opérations à effectuer pour transformer deux chaînes de caractères. La suppression, l'insertion et le remplacement de caractères sont les opérations prises en compte. Par exemple, nous pouvons relever deux opérations, une d'insertion (le caractère "r") et l'autre de remplacement (les caractères "i" "y"), entre les chaînes de caractères "cirrhose" et "cyrhose". Ainsi nous avons E(cirrhose,cyrhose) = 2. Le **String Matching** (noté Str) qui prend en compte la distance d'édition est donné par la formule (1.10) :

$$\textbf{Str(ch1,ch2)} = \max\{0, \frac{min(|ch1|,|ch2|) - E(ch1,ch2)}{min(|ch1|,|ch2|)}\} \in [0,1] \tag{1.10}$$

Dans notre cas, nous avons **Str**(cirrhose,cyrhose) = $\max\{0, \frac{7-2}{7}\} = 0.7$

Les fautes de frappe sont traitées avec les mêmes techniques puisqu'elles sont imprévisibles pour pouvoir être enregistrées à l'avance.

Les variantes de l'ordre des mots représentent l'organisation syntaxique des termes. Elles constituent une autre source de variation. Généralement dans les domaines de recherche d'information (IR) ou d'indexation de documents l'ordre des mots n'est pas significatif et donc ils sont traités comme des sacs de mots, leur structure originale n'est plus considéré et les mots sont triés et traités dans l'ordre alphabétique.

L'appariement devient plus facile et les mesures de similarité pouvant être appliquées peuvent se baser sur le nombre de mots communs (recherche booléenne pondérée). Le défaut principal est d'attribuer le même poids à tous les mots et de privéligier les termes longs (ceux contenant le plus de mots).

D'autres mesures vont calculer la somme des produits TFIDF (qui permet de tenir compte de l'importance de chaque mot). Là encore, on priviligie les longs termes.

La mesure la plus couramment utilisée est celle du Cosine qui, en considérant chaque sac de mots comme étant un vecteur, calcule le cosinus de l'angle entre ces vecteurs. La formule se représentera comme suit :

$$\textbf{COSINE(v1,v2)} : \frac{\sum_{c \in v1 \cap v2} TFIDF(c,v1) * TFIDF(c,v2)}{\sqrt{\sum_{c \in v1} TFIDF(c,v1)^2 * \sum_{c \in v2} TFIDF(c,v2)^2}} \tag{1.11}$$

Avec :

- **c** : un candidat.
- **v1** : le premier vecteur représentant le premier sac de mots (dans notre cas ça peut être un message d'un patient).
- **v2** : le deuxième vecteur représentant le deuxième sac de mots (dans notre cas ça peut être un message d'un médecin).

L'utilisation d'une combinaison linéaire des mesures précédemment citées est recommandée.

Dans certains cas, le mode d'appariement peut différer, comme pour les expansions des sigles où l'abstraction de l'ordre des mots dans les termes peut être cause d'erreurs.

Les mots vides peuvent être source de variation des termes. Dans notre cas les articles et les prépositions sont considérés comme dénués de sens et donc mis de côté. Les mots grammaticaux (articles, adverbes, prépositions, pronoms, *etc.*) sont polysémiques et très fréquents dans les documents, d'où la raison de leur ignorance. La construction d'une liste de mots vides dépend des domaines et des applications.

Les variations morphologiques représentent également une variation de termes comme par exemple pour les deux syntagmes : *hématome anévrismal* et *anévrisme de l'hématome* où les deux mots *anévrisme* et *anévrismal* sont en relation morphologique (suffixation).

L'appariement des termes doit préserver l'équivalence sémantique que ça soit celle obtenue lors d'une affixation [anévrisme vs anévrismal] ou une conversion traitant des mots ayant la même forme graphique, mais dont les catégories syntaxiques et le sens sont différents [muqueuse/Nom vs muqueuse/Adj] ou bien lors d'une supplétion traitant des mots ayant des bases sémantiques équivalentes mais dont les langues d'origine et les formes graphiques sont différentes [estomac/Nom vs gastrique/Adj].

Les variations morphosyntaxiques englobent l'ordre des mots, leurs formes morphologiques et leurs dépendances syntaxiques. Ainsi, leur traitement demande des connaissances syntaxiques des termes comme par exemple pour apparier *sténose de l'aorte* et *aorte sténosée*.

Différentes études se sont basées sur des mesures statistiques afin d'apparier des candidats susceptibles d'être pertinents. On cite Roche et al. [2012] qui présente ses travaux, basés sur l'algorithme PMI-IR (PointWise Mutual Information and Information Retrieval), ayant pour objectif le développement d'une méthode automatique pour traduire des termes spécialisés. PMI-IR consiste à interroger le Web via le moteur de recherche AltaVista pour déterminer des synonymes appropriés. À partir d'un terme donné noté mot, l'objectif de PMI-IR est de choisir un synonyme parmi une liste donnée. Ces choix, notés choix i (i étant un entier), correspondent aux questions du TOEFL (Test of English as a Foreign Language). Ainsi, le but est de calculer, pour chaque mot à traduire, le terme choix i qui donne le meilleur score. Pour ce faire, dans un premier temps, il repère dans les pages retournées par le moteur de recherche certains marqueurs paralinguistiques comme "(", après il extrait ce qu'il y a à l'intérieur, si c'est écrit en Anglais on le rajoute à la liste des choix i. Puis l'algorithme PMI-IR utilise différentes mesures fondées sur la proportion de documents (messages de patients ou de médecins) dans lesquels les deux termes sont présents. Une des mesures couramment utilisée pour calculer une certaine forme de dépendance entre chacun des mots composant une co-occurrence est l'Information Mutuelle :

Information Mutuelle

$$\mathbf{IM(cand1,cand2)} : \frac{nb(cand1,cand2)}{nb(cand1)*nb(cand2)} \tag{1.12}$$

Notons que dans (1.12) nb(a) est égal au nombre de documents contenant le mot "a", et dans le cas d'utilisation de la mesure sur des connaissances exogènes comme le Web par exemple, ça sera égal au nombre de pages retournées par la requête "a" sur un moteur de recherche. nb(cand1,cand2) représentera donc simultanément le cas où "cand1" et "cand2" sont présents dans la même page (on parle de dépendance souple) ou quand ils sont strictement voisins (on parle de dépendance stricte).

Cette mesure peut être adaptée aux co-occurrences ternaires de manière similaire aux travaux de [Jacquemin, 1997]. Ainsi, une extension naturelle consiste à appliquer cette mesure à des syntagmes formés de n mots comme le montre (1.13)

$$\mathbf{IM(x1,...,xn)} : \frac{nb(x1,...,xn)}{nb(x1)*...*nb(xn)} \tag{1.13}$$

Information Mutuelle au Cube

Une mesure qui s'appuie sur l'Information Mutuelle en privilégiant davantage les co-occurrences fréquentes (1.14) :

$$\textbf{IM3(cand1,cand2)} : \frac{nb(cand1,cand2)^3}{nb(cand1)*nb(cand2)} \tag{1.14}$$

Dans certaines définitions on voit apparaître la fonction log2, $log2\frac{nb(exp,cand)^3}{nb(exp)*nb(cand)}$, dont l'utilisation ne change rien puisque c'est une fonction strictement croissante et donc l'ordre des co-occurrences donné par la mesure n'est pas affecté.

Coefficient de Dice

Cette mesure privéligie moins les co-occurrences rares souvent non pertinentes de manière similaire que l'Information Mutuelle.

$$\textbf{Dice(cand1,cand2)} : \frac{2*nb(cand1,cand2)}{nb(cand1)+nb(cand2)} \tag{1.15}$$

Une extension de la formule de Dice (1.15) à n éléments serait :

$$\textbf{Dice(x1,...,xn)} : \frac{n*nb(x1,...,xn)}{nb(x1)+...+nb(xn)} \tag{1.16}$$

Différentes études ont adapté ces mesures au web afin de calculer la similarité entre des candidats, ces derniers pouvant être des mots ou des termes. On cite [Bollegala. 2007.] qui dans sa méthode supervisée, pour deux mots "A" et "B", utilise le moteur de recherche Google en adaptant les mesures *Dice* et *MI* pour le calcul du nombre de pages retournées pour "A", pour "B", et pour "A et B". Il ne se contente pas du nombre de pages retournées mais combine ce résultat avec la fréquence des patrons lexicaux-syntaxiques extraits à partir de fragments de textes retournés par le moteur de recherche pour "A et B". L'avantage est que l'utilisation des fragments nous évite de télécharger les pages web. L'algorithme a été éxécuté pour des paires de noms synonymes et non synonymes issues de la base de données lexicale *WordNet* afin de l'entrainer, les 200 meilleurs patrons exprimant la similarité sémantique ont été pris. Une fois entrainé le modèle pouvait prédire la similarité sémantique entre les deux mots "A" et "B". La non disponibilité des données annotées dans notre cas serait une réalité à ne pas négliger si on veut l'appliquer dans le domaine médical : cela coûterait en temps et en effort afin d'élaborer les données requises pour l'entrainement.

On cite dans ce qui suit deux mesures de similarité sémantique qui comme la plupart des approches récentes utilisent Wikipedia afin de fournir une grande quantité de connaisances structurées du monde sur les termes d'intérêt.

Modèle de vecteur de lien Wikipedia (WLVM)

Ce model permet de définir le degré de connexité de deux articles en se basant uniquement sur la structure des liens hypertextes de Wikipedia.

il s'agit d'abord de créer un vecteur pour les deux articles à comparer

$$x = \left(w(x \to l_1), w(x \to l_2), ..., w(x \to l_n)\right)$$
$$y = \left(w(y \to l_1), w(y \to l_2), ..., w(y \to l_n)\right)$$

Ici, x et y sont les articles à comparer. On répertorie les liens contenus dans x et y, ici représentés par la lettre l suivi d'un indice. On calcule ensuite le poids de chaque lien de x vers li et y vers li, ici représenté par $w(x- > li)$. Les poids respectifs des articles x et y composeront les vecteurs. Le poids d'un lien se calcul de cette manière :

$$w(a \to b) = |a \to b| \times \log\left(\sum_{x=1}^{t} \frac{t}{|x \to b|}\right)$$

Le degré de similitude entre les articles est ainsi défini par l'angle entre les deux vecteurs. A 0° les articles sont parfaitement similaire, à 90° ils n'ont aucune connexité.

Cette méthode est rapide et ne nécessite qu'un dump contenant la structure des liens hypertextes de Wikipédia (assez lèger), là où d'autres méthodes (notamment la notre) nécessite aussi le contenu des articles (beaucoup plus lourd). Cependant, les résultats obtenus ne sont pas aussi précis que d'autres modèles, notammet ESA (que nous aborderons dans la partie suivante).

Analyse sémantique explicite (ESA)

C'est un modèle conçu par Evgeniy Gabrilovich et Shaul Markovitch visant à mesurer le degré de relation entre deux mots ou deux textes en s'appuyant sur wikipedia. Le modèle défini d'abord une liste de termes, aussi appelée 'interpréteur sémantique'. les termes sont associés à une liste de concepts (des articles wikipedia). Ces concepts sont classés selon un poids, calculé par un TF-IDF comme montré dans la figure 4 :

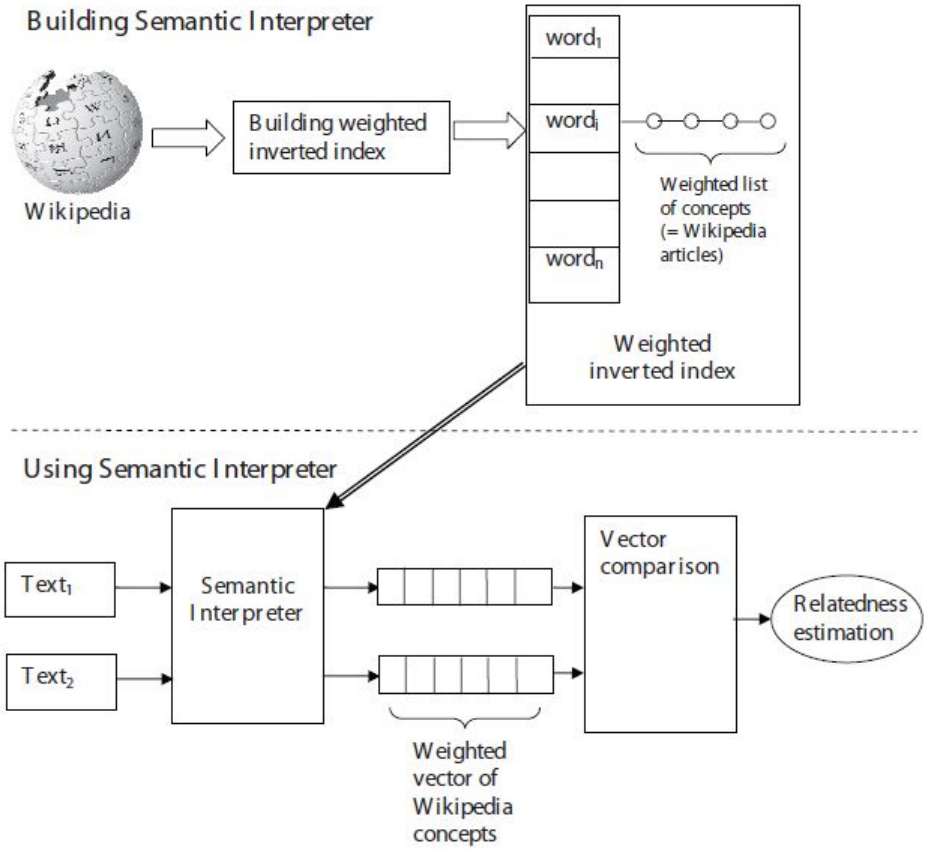

Figure 4 : Construction et utilisation de l'interpréteur sémantique

Lorsque l'on soumet un mot, ou un texte, on procède à un TF-IDF dans le texte, classant ainsi les mots du texte par ordre d'importance. Il s'agit ensuite de récupérer la liste de concepts associée aux mots les plus importants. Ainsi, on défini les grandes lignes du texte soumis. Plus deux mots/textes ont des concepts communs, plus ils sont en relation. Ce modèles s'avère efficace, en effet, le taux de ressemblance avec des résultats trouvés par l'homme s'élève à 0.72 selon Gabrilovich and Markovitch [2007].

Section 2 : Méthodologie

4 Introduction

L'objectif de notre travail est de présenter une méthodologie originale pour extraire des termes biomédicaux souvent utilisés, dans les forums de santé par des patients et de les apparier par la suite à des termes utilisés par les professionnels de santé. Les méthodes d'extraction utilisées dans notre approche prennent en considération diverses approches évoquées dans l'état de l'art. Certaines

sont efficaces pour les fautes d'orthographes, d'autres pour les abréviations, les mots d'argot ainsi que d'autres pour l'appariement sémantique. Nous avons expérimenté ces méthodes sur le forum de santé `http://masantenet.com/`.

Les ressources utilisées sont le thésaurus biomédical de référence pour le français MeSH contenant 27 149 descripteurs répartis dans 16 catégories thématiques différentes (Anatomie, Soins de santé...), l'API Wikipedia : l'encyclopédie universelle, multilingue contenant 1 525 119 articles pour le français et le Diko, le dictionnaire d'associations lexicales contributif et libre de JeuxDeMots `http://jeuxdemots.org/`.

5 Pré-traitement

Nous allons présenter dans la suite différentes méthodes pour apparier des termes utilisés par des patients et des termes utilisés par les professionnels de santé. Chacune de ces méthodes nécessite des pré-traitements spécifiques. Nous listons ci-après l'ensemble de ces prétraitements :

— **0.** Nettoyage initial. Nous avons retiré tout ce qui n'est pas utile pour une analyse syntaxique, par exemple les expressions régulières (ponctuation, caractères spéciaux...). Ainsi nous avons construit une liste des mots vides qui seront supprimés de notre corpus.

— **1.** Suppression des mots vides. Dans les travaux de fouille de texte, il est courant d'éliminer les mots dit vides (*e.g* les prépositions...). Nous n'avons pas appliqué ce prétraitement car nous recherchons toutes les variations des termes candidats, comme par exemple mal de tête, mal à la tête.

— **2.** Étiquetage grammatical. Il existe de nombreux outils comme treetagger Schmid [1994] et brill Brill [1995] pour associer à chaque mot sa fonction grammaticale et éventuellement sa forme lemmatisée. D'après Allauzen and Bonneau-Maynard [2008], treetagger est le plus efficace. Notre choix c'est donc porté sur ce dernier.

— **3.** Uniformisation de la casse. Il s'agit de convertir tous les mots en majuscules ou en minuscules.

— **4.** Corrections orthographiques : les fautes d'orthographe ou de frappe peuvent être corrigées automatiquement. Si ces fautes sont relativement rares dans les textes écrits par les professionnels de santé, elles sont nombreuses dans les textes des non professionnels une correction automatique évitera de créer deux candidats différents pour un mot mal orthographié et sa forme correcte. Sachant qu'on veut garder les termes candidats contenant des erreurs orthographiques et qu'au même temps on vise à diminuer le nombre des candidats dans l'ensemble. On a utilisé Aspell `http://aspell.net/`. qui est un logiciel de correction orthographique utilisant un algorithme de proposition de correction meilleur que ispell Kuenning et al. [2004] : il évalue la prononciation des mots mal orthographiés pour trouver des propositions de corrections plus adaptées. Le danger étant de "corriger" abusivement certains mots corrects mais inconnus du lexique. Nous avons donc concervé deux corpus. Le corpus original contenant des erreurs orthographiques, puis nous avons construit un nouveau corpus corrigé.

6 Approche proposée

Cette section décrit les mesures utilisées, adaptées, et les différentes combinaisons éffectuées selon les types des termes candidats biomédical automatiquement extraits. Dans une première partie, on adapte certaines mesures afin d'extraire et d'apparier des termes patients/médecins, les deux mesures adaptées sont la distance d'édition et la racinisation soit respectivement [levenshtein. 1966.] et [Porter. 1980.]. Dans une deuxième partie on détaille les extentions des mesures de références, plus particulièrement celle de la méthode C-value utilisée et qui dans notre cas prend en considération des patrons linguistiques spécialisés dans le domaine biomédical. On a aussi utilisé TFIDF pour calculer le poids des termes afin d'évaluer leurs importance. L'appariement des termes candidats, les plus pertinents et suivants des patrons linguistiques spécialisés dans le domaine biomédical, patients et médecins s'effectue en utilisant une approche s'appuyant sur deux modèles : Modèle de vecteur de lien Wikipedia **WLVM** Milne [2007] et Analyse sémantique explicite **ESA** Gabrilovich and Markovitch [2007].

Notre approche se constitue de 6 étapes précédées d'une étape d'étiquetage grammatical :

— **0.** Etiquetage grammatical du corpus

— **1.** Adaptation de la distance de levenshtein pour l'extraction et l'appariement approximatif de chaînes de caractères (celles contenant des fautes d'orthographe)

— **2.** Adaptation de l'algorithme de Porter pour l'extraction et l'appariement de chaînes de caractères (celles contenant des abbréviations)

— **3.** Extraction des mots clés et classification des termes candidats

— **4.** Extraction des termes candidats suivants des patrons

— **5.** Appariement des termes candidats patients/médecins sémantiquement similaires (mots d'argot)

ETAPE 0 : Etiquetage grammatical

L'étiquetage grammatical (en Anglais Part of Speech "POS") associe chaque mot à sa catégorie grammaticale (nom, adjectif). Ce processus est basé sur la définition du mot ou sur le contexte dans lequel il apparaît. Cette étape a été effectuée sur les deux corpus : celui contenant les messages des médecins et celui contenant les messages des patients ce dernier a été pris tel quel dans un premier temps puis corrigé avec le correcteur Aspell dans un deuxième temps. Dans cette étape, comme suggéré par la méthode C-value, l'étiquetage a été appliqué sur la totalité des deux sous corpus. On a évalué deux outils (TreeTagger et Brill) et finalement choisi TreeTagger qui donne de meilleurs résultats.

La figure 5 décrit l'approche proposée. Nous détaillons dans la suite les 5 étapes principales de ce modèle :

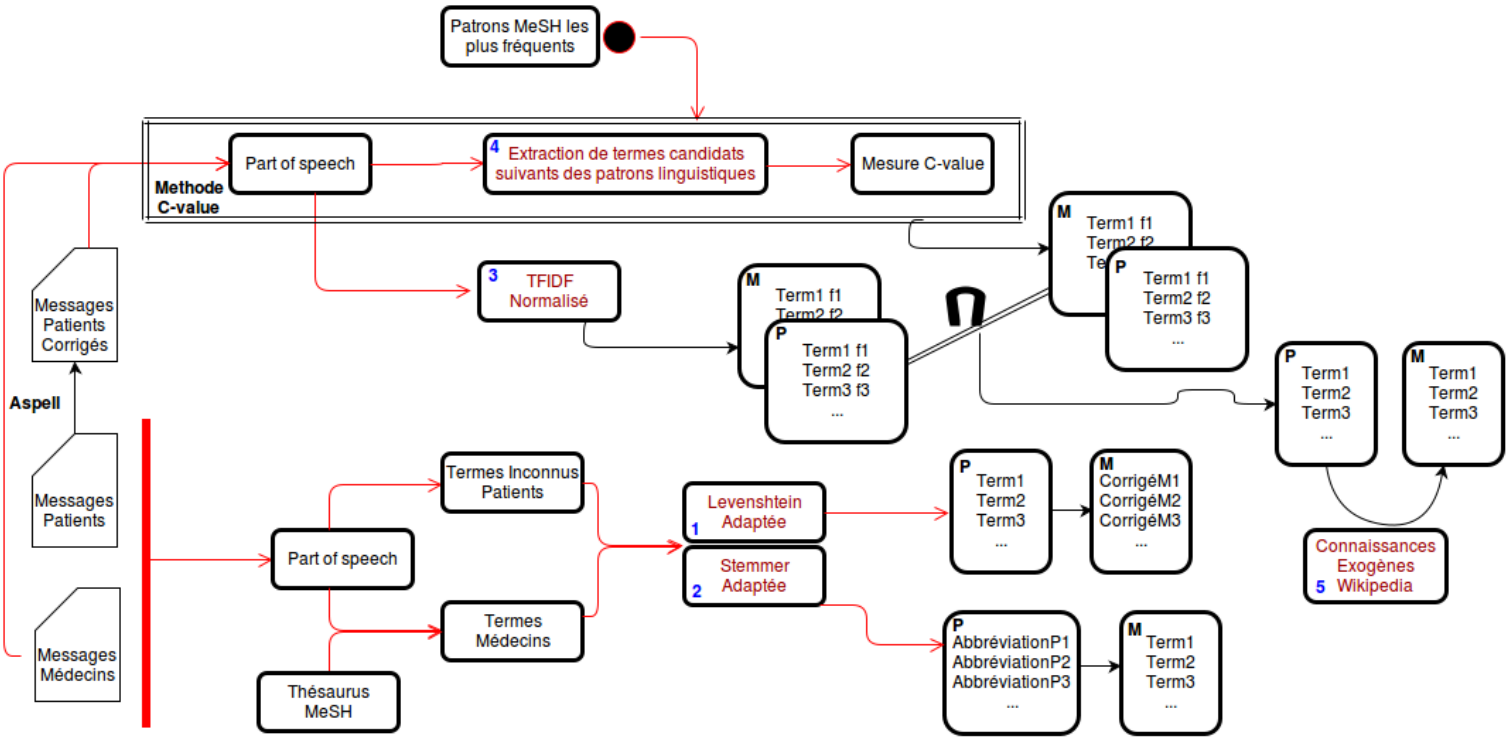

Figure 5 : Approche pour l'extraction et l'appariement des termes candidats

ETAPE 1 : Adaptation de la distance de Levenshtein

La distance de Levenshtein mesure la similarité entre deux chaînes de caractères. Elle est égale au nombre minimal de caractères qu'il faut supprimer, insérer ou remplacer pour passer d'une chaîne à l'autre.

Son nom provient de Vladimir Levenshtein qui l'a définie en 1965. Elle est aussi connue sous le nom de *distance d'édition*. Elle est égale au nombre minimal de caractères qu'il faut supprimer, insérer ou remplacer pour passer d'une chaîne à une autre. Cette distance est d'autant plus grande que le nombre de différences entre les deux chaînes est grand. Nous avons adapté cette mesure pour prendre en considération essentiellement les fautes d'orthographe très courantes chez les patients mais également présentes chez les professionnels de santé.

Principe de l'adaptation Classiquement, on définit la distance de Levenshtein entre deux mots M1 et M2, comme le coût minimal pour aller du mot M1 à M2 en effectuant les trois opérations élémentaires suivantes :

— **0.** substitution d'un caractère de M en un caractère de P.

— **1.** ajout dans M d'un caractère de P.

— **2.** suppression d'un caractère de M.

Pour notre application, nous avons choisi d'associer un coût de 1 pour chacune de ces opérations, sauf dans le cas d'une substitution de caractères « semblables » (*e.g* variation d'accent) pour lequel nous avons donné un coût de 0.5. Nous avons également posé 4 conditions sur l'appariement des mots : **1)** ils doivent commencer avec la même lettre ; **2)** ils doivent être composés de plus de trois caractères ; **3)** la comparaison doit être insensible à la casse ; **4)** la distance maximum considérée est de 2.

Application dans le processus général Comme montré sur la figure 8, la distance adaptée a été utilisée après le pré-traitement du sous-corpus contenant les messages des patients (voir *ETAPE 0*) sur les noms, adjectifs et verbes non reconnus par l'étiqueteur grammatical et non présents dans le thésaurus MeSH. Nous en déduisons un ensemble de termes, les Termes Inconnus Patients, incluant les mal orthographiés que l'on compare avec les Termes Médecins grâce à l'ETAPE 1 Levenshtein adaptée. Nous obtenons en sortie des termes patient (Term1) appariés à des termes corrigés médecins (CorrigéM1).

Exemple : le terme *cyrhose* apparaissant dans l'ensemble des Termes Inconnus Patients est apparié avec le terme *cirrhose* apparaissant dans Termes Médecins.

ETAPE 2 : Adaptation de l'algorithme de Porter

Dans les langues flexionnelles ou agglutinantes comme le français, les termes subissent des flexions. Une flexion est une modification morphologique affectant un terme pour marquer sa position grammaticale, le temps de conjugaison *etc.* Par exemple, le verbe « manger » se fléchit en « mangeons » lorsqu'il est placé dans une phrase au présent et a pour sujet une première personne du pluriel. Le mot cheval se fléchit en chevaux au pluriel. L'objectif de la lemmatisation (normalisation) est de retrouver la forme canonique commune à plusieurs mots fléchis.

Il existe deux familles de lemmatiseurs : **1)** les lemmatiseurs algorithmiques sont souvent plus rapides et permettent d'extraire des racines à partir de mots même inconnus. Leur taux d'erreur est par contre plus élevé, car ils peuvent grouper des mots qui ne devraient pas l'être (sur-racinisation, ou over-stemming) ; **2)** les lemmatiseurs par dictionnaire ne font par contre pas d'erreur sur les mots connus mais ne fonctionnent pas sur les mots inconnus. Ils sont aussi plus lents et nécessitent malgré tout la suppression de suffixes avant toute recherche de la racine correspondante dans le dictionnaire.

Principe de l'algorithme adapté L'algorithme de Porter que nous avons utilisé est un algorithme de lemmatisation algorithmique Porter [1980]. Cet algorithme est très utilisé pour la langue anglaise, mais son efficacité est limitée pour la langue française où les flexions sont plus importantes et diverses. Il reste toutefois un algorithme fondamental couramment utilisé en TALN.

Nous nous sommes basé sur Paternostre et al. [2002] qui décrit une adaptation de l'algorithme de désufixation de PORTER pour le français.

Cet algorithme réalise un pseudo découpage. Il travaille sur les mots et mots composés à leur plus bas niveau de lettres. Pour cela, il partitionne l'ensemble des lettres de l'alphabet latin en deux classes :

— les voyelles (notées v) : a, à, â, e, è, é, ê, ë, i, î, ï, o, ô, u, û, ù et y quand il est précédé d'une consonne.

— les consonnes (notées c) : toutes les autres lettres et Y quand il est précédé d'une voyelle.

D'après cette classification des lettres, chaque mot peut s'écrire [C]VC...[V] avec V une séquence d'au moins une voyelle et C une séquence d'au moins une consonne. Les couples VC correspondent alors aux pseudo-syllables du mot. Le nombre de pseudo-syllables du mot est appelée la **mesure du mot** et se note **m**.

L'algorithme utilise également des règles de transformation morphologiques ayant la syntaxe suivante : **(<condition>) S1 -> S2**, avec :

— **S1** est le suffixe que l'on retire au mot afin d'obtenir le radical ;

— **S2** est le suffixe que l'on ajoute au radical si la condition est remplie ;

— **condition** est une condition que le radical doit vérifier pour que la règle soit appliquée.

La condition s'applique sur le mot considéré privé de S1, partie du mot qu'on appellera radical par la suite. Il y a un certain nombre de notations spécifiques à certaines conditions :

— *d signifie que le radical se termine par deux consonnes.

— *S signifie que le radical se termine par la lettre S.

— *v* signifie que le radical contient une voyelle.

Principe de l'adaptation Nous avons ajouté un ensemble de suffixes les plus utilisés dans le domaine biomédical aux règles d'extraction de racines. En appendice se trouve l'ensemble des règles que nous appliquons pour l'étude de la morphologie du français. Ainsi après l'application de la règle -e > - *chienne* deviendra *chienn*. Ensuite, en phase trois et suivant la règle -nn > n, *chienn* deviendra *chien*. un exemple de nouvelle règle est l'application de -logue > -, *gynécologue* deviendra *gynéco*.

Application dans le processus général Comme montré sur la figure 8, l'algorithme de Porter adapté a été utilisé après le pré-traitement du sous-corpus contenant les messages des patients et des médecins (voir *ETAPE 0*) sur les noms, adjectifs et verbes non reconnus par l'étiqueteur grammatical et non présents dans le thésaurus MeSH. Cette adaptation ne permet pas seulement de réduire le nombre de termes dans nos deux sous-corpus médecins et patients mais notamment d'extraire certaines abréviations souvent utilisées par les patients et de les apparier avec les termes utilisés par les médecins. Nous obtenons en sortie des termes patient (AbbréviationP1) appariés à des termes médecins (Term1).

Exemple : le terme oncologue utilisé par les médecins est ainsi apparié avec le terme onco utilisé par les patients.

ETAPE 3 : Classification des termes candidats

Principe de la pondération Nous considérons ici que chaque message constitue un document appartenant à une des deux collections (corpus des patients ou des médecins). Nous avons appliqué la mesure tf-idf qui nous a permis d'évaluer l'importance d'un terme contenu dans un document, relativement à une collection (voir formule dans la section 2.4). Le poids augmente proportionnellement avec le nombre d'occurrences du mot dans le document. Il varie également en fonction de la fréquence du mot dans la collection. Ainsi, la fréquence inverse du document (idf) mesure l'importance du terme dans l'ensemble des documents. Dans la formule du tf-idf, le idf donne un poids plus important aux termes les moins fréquents, considérés comme les plus discriminants. L'intuition est ici de capturer des termes qui sont utilisés fréquemment par les patients mais qui ne sont pas utilisés par les professionnels de santé. La mesure tf-idf est calculée avec un nombre variable d'éléments et les valeurs obtenues sont donc hétérogènes. Afin de manipuler ces listes de résultats, les poids obtenus de chaque document sont normalisés pour la totalité du corpus.

Application dans le processus général Comme montré sur la figure 8, après correction avec *Aspell* et pré-traitement des deux sous-corpus, le texte a été lemmatisé (regrouper les mots d'une même famille) et segmenté, la pondération a été effectuée. Pour implémenter cette étape, nous avons généré un fichier *Arff* qu'on a fournit au logiciel Weka `http://www.cs.waikato.ac.nz/ml/weka/`. Ainsi nous avons utilisé les algorithmes de classification fournis par Weka pour classifier nos termes candidats patients et médecins. À l'issue de cette étape nous obtenons une liste de termes non appariés.

Exemple : On trouve parmi les termes bien classifiés *mal de tête* et *cancer du sein* respectivement dans le CP et CM.

ETAPE 4 : Extraction des termes candidats suivants des patrons L'intuition ici est que les termes médicaux ont une structure syntaxique similaire qu'ils soient utilisés par des professionnels de santé ou des patients. La méthode proposée repose sur deux étapes : l'extraction des patrons puis l'utilisation de ces patrons pour l'extraction des candidats.

Principe de l'extraction des patrons En nous inspirant des travaux de Ventura et al. [2013] nous avons construit une liste de patrons lexicaux les plus communs suivant la structure syntaxique des termes existants dans la base de données biomédicale MeSH. Pour cela, nous réalisons un étiquetage des termes biomédicaux avec TreeTagger, puis nous calculons la fréquence des structures syntaxiques et sélectionnons les 200 meilleurs comme patrons. 65 000 termes issus du MeSH sont utilisés pour construire la liste des patrons. Le Tableau 3 présente des exemples de patrons triés par fréquence.

	Patrons
1	Noun
2	Noun Adj
3	Noun Prep Noun
4	Noun Adj Adj
5	Noun Prep :det Noun
6	Noun Prep ProperNoun
7	Noun ProperNoun
8	Noun Noun
9	Noun Prep Noun Adj

Tableau 3 : Exemples des 9 patrons les plus fréquents

Principe de l'utilisation des patrons pour l'extraction des candidats Nous filtrons le corpus patient et médecin en utilisant les patrons calculés précédemment pour sélectionner les termes ayant une structure syntaxique appartenant à la liste de patrons.

Application dans le processus général Comme montré sur la figure 8, après correction avec *Aspell* et pré-traitement des deux sous-corpus, chaque sous-corpus est soumis indépendament, puisque la méthode C-value prend un seul document texte comme entrée. La mesure statistique C-value (voir section 2.4) est appliquée afin de calculer la force de l'association d'un terme aux concepts du domaine dans le but d'améliorer l'extraction imbriquée. À l'issue de cette étape, nous obtenons deux listes (patients et médecins) de termes candidats qui respectent les patrons linguistiques et classés par ordre d'importance.

Exemple : On trouve parmi les termes bien classifiés *mal de tête* et *cancer du sein* respectivement dans le CP et CM.

ETAPE 5 : Alignement des termes patient et médecins

Plusieurs méthodes d'alignements ont été envisagés pour les termes candidats issues des étapes 3 et 4 et sont décrites ci-après.

Alignement 1 basé sur l'organisation des messages (question/réponse) Cette première méthode est basée sur la supposition que les médecins reformulent les messages des patients avant de répondre à leur question. Nous avons donc considéré des messages appariés (question d'un patient *vs.* réponse d'un médecin). Pour un terme patient pertinent (selon sa valeur tfidf), on génére un sous corpus construit des réponses médecin dont les questions patient contiennent ce terme. Nous calculons les termes pertinents dans ce sous corpus et que nous supposions intéressant à apparier avec le terme candidat patient. Or, nous nous sommes rendu rapidement compte que les associations extraites étaient plutôt du type symptôme (patient) - traitement (médecin). Par exemple, le résultat

obtenu pour le terme candidat *mal de tête* selon ce processus (15 messages réponses à des messages questions contenant le terme) est : *vitamine C*, *repos*.

Alignement 2 basé sur Wikipedia L'idée d'utiliser la ressource Wikipedia était une parmi les solutions envisageable, et puisque pendant mon TER M1 intitulé *Un programme qui joue à jeuxdemots* encadré par Monsieur *Lafourcade Mathieu* j'ai pu intéragir avec cette dernière. C'était logique de s'orienter vers ce sens. L'approche que l'on propose ici s'appuie sur les deux modèles : WLVM et ESA. Nous avons envisagé 3 versions de cette approche.

Dans la première version, pour un mot soumis (*e.g* 'A'), nous récupérons le contenu de la page wikipedia associée à ce mot. Nous traitons cette page en récupérant tous les liens *internes* (ceux restant dans le domaine de nom wikipedia). Ces liens pointent des pages contenant des mots associé à 'A'. Nous parcourons ensuite chaque lien pour récupérer le mot associés et les urls contenues dans la page. Si parmi ces urls se trouve l'url de la page de 'A', le mot correspondant est compté comme étant associé. Bien que efficace, cette version est très lente à cause de l'ouverture d'un très grand nombre de liens.

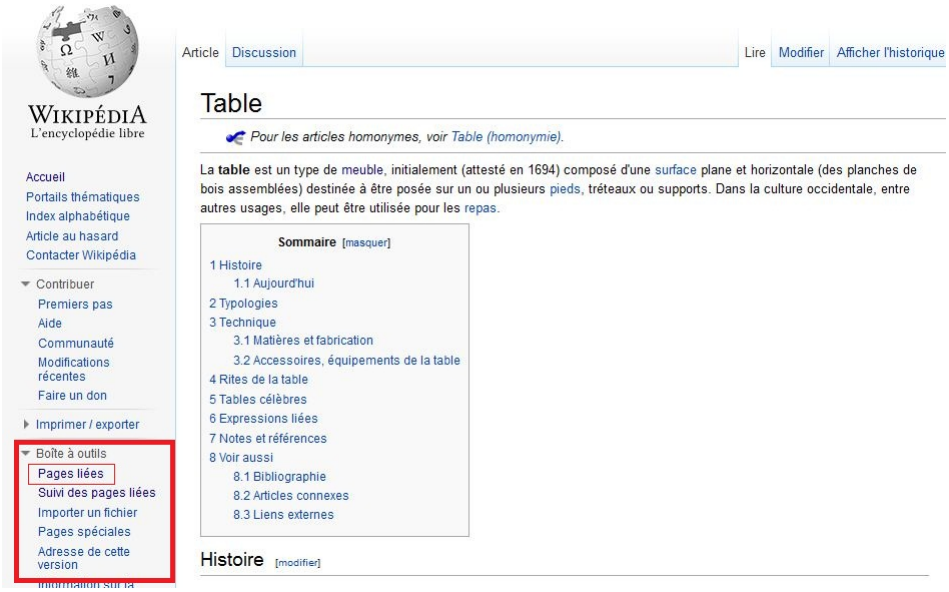

Figure 6 : Option pages liées

Dans la deuxième version, nous avons exploité la notion de *pages liées* située dans la catégorie *boîte à outils* (présente dans n'importe quel article Wikipedia comme le montre la *figure 6*). Cette boîte contient un lien vers une page répertoriant tous les mots (et les liens vers ces mots) pointant vers le mot soumis (ici, toujours appelé 'A'). Il suffit alors de comparer les liens de la page de 'A' avec ceux de la page des mots liés. Un mot ayant un lien pointant vers les deux pages sera considéré comme lié a 'A'. Dans la *figure 7*, 'B', 'C' et 'D' ont un lien réciproque avec 'A' et nous les considérons liés, contrairement à 'E' et 'F' qui ont respectivement un lien entrant et sortant avec 'A'. 'G' n'est pas du tout lié à 'A'.

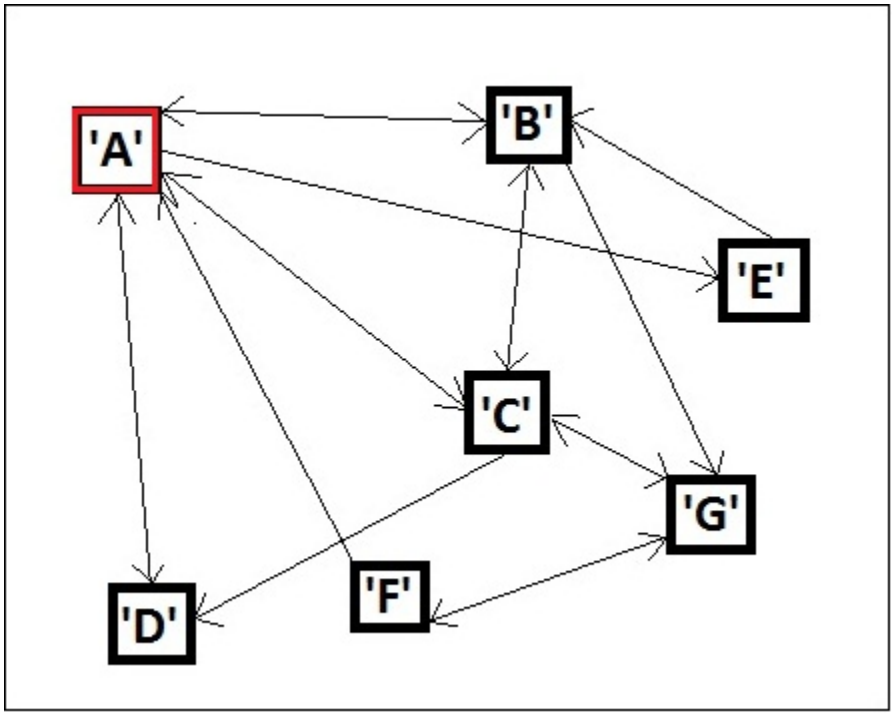

Figure 7 : Exemple de types de liaison

La troisième version fait appel à l'API de wikipedia. Cette API permet d'obtenir la liste des mots ayant un lien vers un mot cible et la liste des mots pointés par le mot cible. On obtient ainsi rapidement une liste de termes exploitables. Trois nouvelles notions ont été utilisées pour affiner les résultats. **1)** nous supprimons les termes issus de pages répertoriant les homonymes, les pages correspondantes à une année *etc.* **2)** nous scorons les termes. Soit A le mot d'origine et B le mot candidat. Nous comptons tout d'abord le nombre d'occurrences de 'B' dans l'article de 'A', puis réciproquement de 'A' dans 'B'. Le Score est ainsi incrémenté le nombre de fois que l'on compte d'occurrences dans les deux calculs. Nous comptons également le nombre d'occurrences de 'B' apparaît dans la liste des mots liés de chaque mots considérés comme liés à 'A'. Si 'B' et 'C' sont liés à 'A' et que 'B' apparaît dans la liste des mots liés à 'C', alors le score de 'B' est incrémenté de 1. Une fois le score calculé pour chaque terme lié, les résultats sont triés par ordre décroissant de score.

Remarque L'API Wikipédia est sensible à la casse et aux intitulés choisis pour décrire les pages. Par exemple, elle reconnaît 'Migraine' mais pas 'migraine' et elle reconnaît 'Dépression (psychiatrie)' mais pas 'dépression'. Nous avons donc implémenté une fonction qui interroge l'API Wikipedia pour nous donner le terme le plus proche qu'elle aura trouvé par rapport à un terme source. Dans l'exemple précédent, notre fonction retourne 'Dépression (psychiatrie)' si nous entrons 'dépression'.

Application dans le processus général Comme montré sur la figure 8, les résultats obtenus des étapes 3 et 4 sont associés afin de construire deux listes se constituant respectivement des

termes candidats patient *f1* et des termes candidats médecin *f2* les plus fréquents et qui suivent des patrons linguistiques. Dans cette étape on va lire le fichier qui contient les termes médecin *f2* et pour chaque terme on va appliquer le processus qui suit la troisième version de notre programme. Cela nous générera un ensemble de termes liés à ce terme candidat *Ens1*. On utilisera le fichier qui contient les termes patient *f1* comme filtre et donc parmi les résultats obtenus dans *Ens1* on ne va garder (stocker dans une base de données) que ceux qui apparaissent dans *f1*. Ainsi, on continue jusqu'à l'appariement de tout les termes dans *f1* avec ceux dans *f2* selon les liens générés par notre processus se basant sur Wikipedia.

À l'issue de cette étape, nous aurons stocké dans notre base de données des paires (terme médecin - terme patient) triés par ordre décroissant de score.

Exemple : Si dans le *f2* existe le terme *Cancer*, on va appliquer notre processus qui va nous générer un ensemble de termes liés à *Cancer*. On va trouver : *tumeur*, *Alcool*, *Crabe*, *néoplasme*, *etc.* alors en utilisant le filtre *f1* le terme *Cancer* dit par les médecins sera apparié aux termes *tumeur*, *Alcool*, *Crabe*, *etc.* mais pas à *néoplasme* car ce dernier terme n'est pas employé par les patients.

La figure 8 présente le diagramme de classe de notre programme :

Diagramme des classes

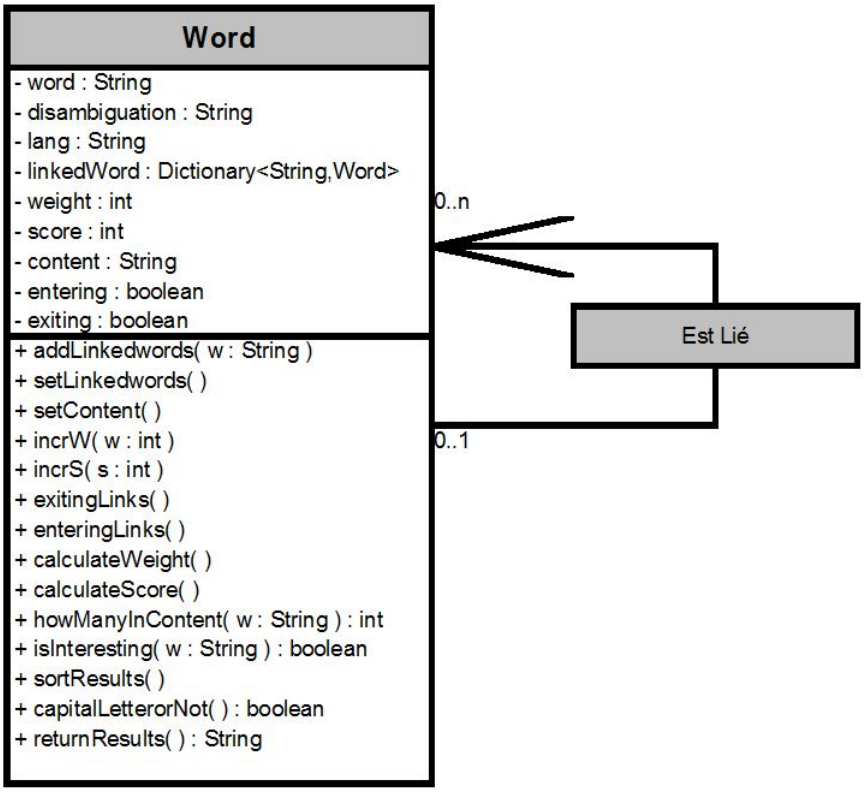

Figure 8 : Diagramme de classe

6.1 Exemple de résultat

La figure 9 présente un exemple de résultat obtenu avec notre approche sur un ensemble de candidats extraits du site masantenet `http://masantenet.com/`.

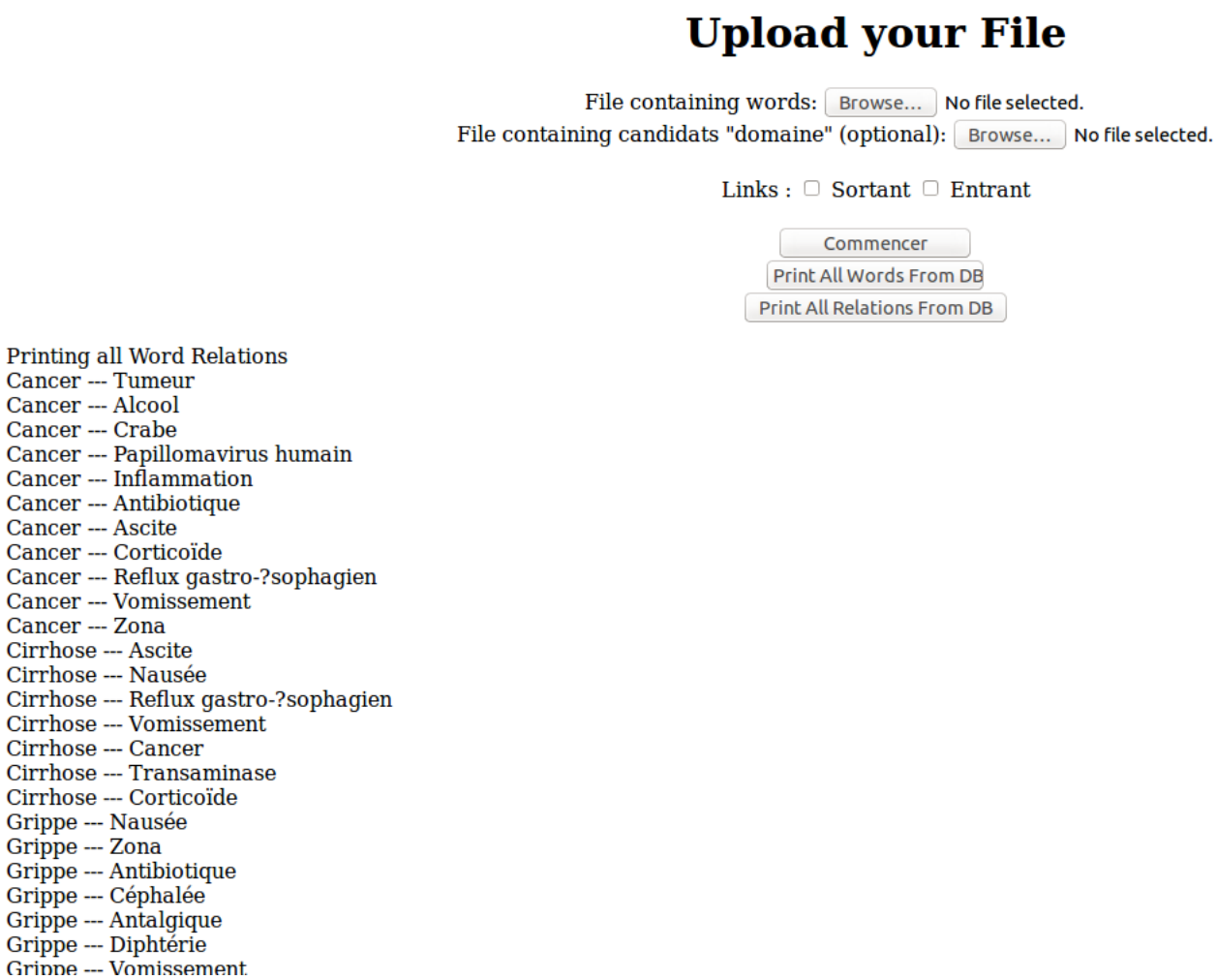

Figure 9 : Exemple de résultat obtenu

Section 3 : Données et Expérimentations

Nous avons utilisé comme corpus une collection de paires de question/réponse disponibles en ligne sur le site masantenet `http://masantenet.com/`. Notre corpus extrait se contitue de 6 534 messages (soit 557 790 mots)

7 Validation des données

Les étapes 1 et 2 de notre approche (voir Section 2) nous ont permis d'apparier un nombre important de termes. Nous avons évalué certains de ces résultats manuellement.

Pour l'Etape 1, nous avons pris 100 paires (terme patient - terme médecin) générés et avons obtenu 92 correspondant vraiment à des erreurs commises par les patients et leurs corrigés coté médecin. Soit une précision de 0,92. Ceci s'explique par l'efficacité de la méthode (Lenveshtein adaptée) appliquée et le nombre important d'erreurs comises par les patients.

La figure 10 représente certains résultats annotés :

Figure 10 : Résultats ETAPE 1 annotés

L'Etape 2 nous a permis de générer une centaine (216) de paires (terme patient - terme médecin) correspondant vraiment à des abréviations utilisées par les patients et leurs termes associés coté médecin. Cette méthode ayant pour but principal la réduction du nombre de candidat. Grâce aux régles rajoutées on a pu apparier certains termes. On cite par exemple :

— rhin vs rhinite

— onco vs oncologue

— gynéco vs gynécologie

— urétr vs urétrite

— gloss vs glossite

Pour l'ETAPE 5 nous avons établi 2 types de validation :

7.1 Validation Automatique

Dans le but de valider automatiquement les résultats obtenus suite à l'appariement des termes candidats sémantiquement similaires. Nous avons procèdé par vérification de l'existance des paires (terme patient - terme médecin) dans le Dico de `http://www.jeuxdemots.fr/` sachant que ce dernier rassemble 112 types de relations dont 179 578 occurences de la relation synonymie, 420 de la relation synonymie strict et 345 185 d'occurences de la relation isA, *etc.* Nos résultats sont évalués en terme de précision. Notons comme conséquence que le rappel est égal à 100% avec la totalité des termes extraits.

Nous nous sommes interessés à une certaine catégorie de relations pour détecter si un couple de termes est lié d'une manière ou d'une autre. Dans l'annexe (Section 5) se trouve la liste des relations utilisées.

Nous avons developpé un script en python en utilisant la bibliothèque BeautifulSoup `http://www.crummy.com/software.BeautifulSoup/` qui permet d'extraire du contenu html dans le but de lancer une requête pour chaque couple de termes afin de vérifier l'existence d'une quelconque relation.

La figure 11 représente certains résultats annotés automatiquement avec le Dico :

```
Abcès --- Kyste --- associationsIdees synonymes
Abcès --- Inflammation --- associationsIdees isA
Abcès --- Infection --- associationsIdees isA
Acné --- Bouton --- associationsIdees synonymes
Addiction --- Dépendance --- associationsIdees synonymes consquence
Allergie --- Allergène --- associationsIdees family
Allergie --- Pollen --- associationsIdees
Angine --- Mal de gorge --- associationsIdees synonymes
Angiome --- Hémangiome --- associationsIdees hyponyme
Angiome --- Tumeur --- associationsIdees synonymes
Anxiété --- Peur --- associationsIdees synonymes hyponyme sentiment
Anxiété --- Angoisse --- associationsIdees synonymes
Anxiété --- Stress --- associationsIdees synonymes
Anxiété --- Trac --- associationsIdees
Anxiété --- Inquiétude --- associationsIdees synonymes sentiment
Bec --- Bouche --- associationsIdees synonymes
Biopsie --- Ponction --- associationsIdees synonymes
Cancer --- Maladie --- associationsIdees raffinementSemantique Domaine isA magn
Cancer --- Leucémie --- synonymes hyponyme
Cancer --- Métastase --- associationsIdees synonymes
Cancer --- Crabe --- associationsIdees synonymes family
Cancer de la peau --- Mélanome --- associationsIdees
Cancer du sein --- Tumeur --- associationsIdees isA
Cancer du sein --- Femme --- cible
Cancer du sein --- Biopsie --- diagnostique
Comprimé --- Médicament --- associationsIdees raffinementSemantique isA
Comprimé --- Cachet --- associationsIdees synonymes isA
Cordon ombilical --- Cordon --- associationsIdees synonymes
Cytoplasme --- Membrane --- associationsIdees hasPart
Cytoplasme --- Cellule --- associationsIdees synonymes holonyme
Démangeaison --- Prurit --- associationsIdees synonymes
Démangeaison --- Allergie --- associationsIdees
Diarrhée --- Gastro-entérite --- associationsIdees synonymes
Éjaculation précoce --- Éjaculation --- associationsIdees isA
Grippe --- Virus --- associationsIdees raffinementSemantique isA holonyme
Grippe --- Maladie --- associationsIdees raffinementSemantique Domaine isA
Grippe --- Vaccination --- associationsIdees
Grippe --- Épidémie --- associationsIdees holonyme
```

Figure 11 : Résultats ETAPE 5 annotation automatique

Nous avons généré 6 320 paires de termes par notre programme (voir ETAPE 5). Grâce à cette validation automatique nous avons pu valider 1 382 couples. Soit 26% des résultats.

7.2 Validation Manuelle

Afin d'avoir la précision réelle, parce que dans la validation manuelle il y a des termes qui n'existe pas dans notre **Dico** puisque ce dernier n'est pas spécialisé dans le domaine médical. Alors, nous avons exploré une partie des paires de termes restantes afin de les valider manuellement. Pour cela, on a utilisé trois types de relations : *AssociationDirecte*, *AssociationIndirecte*, *Loin*. Deux termes étant en *association directe* aura pour but d'augmenter la précision de notre programme, alors que si les deux termes étant *loin* cela diminuera la précision.

Le tableau ci-dessous montre le nombre de paires de termes validées manuellement :

	90 paires	180 paires
AssociationDirecte	35	75
AssociationIndirecte	46	86
Loin	9	19

Tableau 4 : Nombre de couples validés manuellement

On remarque que les termes considérés par l'expert comme étant directement associés étant toujours le plus élevé et que le reste des couples sont indirectement associés. Le nombre de paires de termes considéré Loin étant faible, cela vient confirmer la validité de notre approche.

Section 4 : Conclusion et perspectives

En conclusion, lors de ce stage nous avons proposé une approche originale pour apparier des termes candidats patient et médecin. Les types des termes candidats étant hétérogènes nous a incité à décomposer notre approche en différentes étapes.

Nous avons apparié d'abord les termes mal orthographiés utilisés par les patients avec leurs corrections situées dans les réponses des médecins.

Nous avons ensuite adapté et implémenté une méthode de désufixation afin d'apparier certaines abbréviations souvent utilisées par les patients avec les termes exacts correspondant utilisés par les médecins

Nous avons ensuite généré un ensemble de termes (mot, bigramme, trigramme) les plus fréquents dans notre corpus en se basant sur des mesures statistiques et d'autres respectant des patrons linguistique.

Nous avons essayé d'aligner nos termes candidats en se basant sur l'organisation des messages (question/réponse). Les premiers retours nous ont permis de partiellement invalider notre hypothèse de départ sur la possibilité d'apparier des termes patient/médecin en alignant les messages. En effet cette méthode doit être expérimentée sur un jeu de données plus large et représentatif que celui réalisé.

Nous avons également essayé d'aligner nos termes candidats en implémentant une méthode basée sur des connaisances exogènes (Wikipedia).

Nous avons proposé des pistes pour annoter nos résultats. Une validation automatique a été implémentée en exploitant les informations du dictionnaire d'associations lexicales contributif et libre de JeuxDeMots `http://jeuxdemots.org/`. Puis une autre validation manuelle a été établis grâce à la collaboration avec l'expert.

En résumé ce travail était axé sur l'analyse de la structure des messages publiés dans les forums de santé afin de pouvoir extraire dans un premier temps des termes pertinents du domaine utilisés par les patients et ceux utilisés par les médecins et de les associer dans un second temps. En exploitant des connaissances endogènes et d'autres exogènes. Le but étant de faciliter la communication entre les deux parties (patients et médecins).

Section 5 : Annexes

Relations utilisées pour la validation automatique *Dico*

- associationsIdees
- raffinementSemantique
- raffinementMorphologique
- Domaine
- synonymes
- isA
- hyponyme
- hasPart
- holonyme
- caract
- data
- magn
- family
- consquence
- against
- domaine1
- instance
- similar
- synStricte
- cible
- symptome
- diagnostique
- sentiment

Règles appliquées dans l'algorithme de désufixation

- "esre1>", // -erse > -ers
- "esio1>", // -oise > -ois
- "siol1.", // -lois > -loi
- "siof0.", // -fois > -fois
- "sioe0.", // -eois > -eois
- "sio3>", // -ois > -
- "st1>", // -ts > -t
- "sf1>", // -fs > -f
- "sle1>", // -els > -el
- "slo1>", // -ols > -ol
- "sé1>", // -és > -é
- "étuae5.", // -eauté > -
- "eugol5.", // -logue > -
- "eigol5.", // -logie > -
- "eisatce7.", // -ectasie > -
- "eihparg7.", // -graphie > -
- "étuae2.", // -eauté > -eau
- "tnia0.", // -aint > -aint

— "tniv1.", // -vint > -vin
— "tni3>", // -int > -
— "suor1.", // -rous > -ou
— "suo0.", // -ous > -ous
— "sdrail5.", // -liards > -l
— "sdrai4.", // -iards > -i
— "erèi1>", // -ière > -ier
— "sesue3x>", // -euses > -euse
— "esuey5i.", // -yeuse > -i
— "esue2x>", // -euse > -eux
— "se1>", // -es > -e
— "erèg3.", // -gère > -g
— "eca1>", // -ace > -ac
— "esiah0.", // -haise > -
— "esi1>", // -ise > -is
— "siss2.", // -ssis > -ss
— "sir2>", // -ris > -r
— "sit2>", // -tis > -t
— "egané1.", // -énage > -énag
— "egalli6>", // -illage > -
— "egass1.", // -ssage > -sag
— "egas0.", // -sage > -
— "egat3.", // -tage > -
— "ega3>", // -age > -
— "ette4>", // -ette > -
— "ett2>", // -tte > -t
— "etio1.", // -oite > -oit
— "tioç4c.", // -çoit > -c
— "tio0.", // -oit > -oit
— "et1>", // -te > -t
— "eb1>", // -be > -b
— "snia1>", // -ains > -ain
— "eniatnau8>", // -uantaine > -
— "eniatn4.", // -ntaine > -nt
— "enia1>", // -aine > -ain
— "niatnio3.", // -ointain > -oint
— "niatg3.", // -gtain > -gt
— "eé1>", // -ée > -é
— "i1>", // -i > -
— "eti3>", // -ite > -
— "sid2.", // -dis > -d
— "sic2.", // -cis > -c
— "esoi4.", // -iose > -
— "ed1.", // -de > -d
— "ai2>", // -ia > -
— "a1>", // -a > -

— "adr1.", // -rda > -rd
— "tnerè5>", // -èrent > -
— "evir1.", // -rive > -riv
— "evio4>", // -oive > -
— "evi3.", // -ive > -
— "fita4.", // -atif > -
— "fi2>", // -if > -
— "enie1.", // -eine > -ein
— "sare4>", // -eras > -
— "sari4>", // -iras > -
— "sard3.", // -dras > -d
— "sart2>", // -tras > -tr
— "sa2.", // -as > -
— "tnessa6>", // -assent > -
— "tnessu6>", // -ussent > -
— "tnegna3.", // -angent > -ang
— "tnegi3.", // -igent > -ig
— "tneg0.", // -gent > -gent
— "tneru5>", // -urent > -
— "tnemg0.", // -gment > -gment
— "tnerni4.", // -inrent > -in
— "tneiv1.", // -vient > -vien
— "tne3>", // -ent > -
— "une1.", // -enu > -en
— "en1>", // -ne > -n
— "nitn2.", // -ntin > -
— "ecnay5i.", // -yance > -i
— "ecnal1.", // -lance > -lanc
— "ecna4.", // -ance > -
— "ec1>", // -ce > -c
— "nn1.", // -nn > -n
— "rit2>", // -tir > -
— "rut2>", // -tur > -t
— "rud2.", // -dur > -d
— "ugn1>", // -ngu > -ng
— "eg1>", // -ge > -g
— "tuo0.", // -out > -out
— "tul2>", // -lut > -l
— "tû2>", // -ût > -
— "ev1>", // -ve > -v
— "vè2ve>", // -èv > -ev
— "rtt1>", // -ttr > -tt
— "emissi6.", // -issime > -
— "em1.", // -me > -m
— "ehc1.", // -che > -ch
— "céi2cè.", // -iéc > -ièc

— "libi2l.", // -ibil > -ibl
— "llie1.", // -eill > -eil
— "liei4i.", // -ieil > -i
— "xuev1.", // -veux > -veu
— "xuey4i.", // -yeux > -i
— "xueni5>", // -ineux > -
— "xuell4.", // -lleux > -l
— "xuere5.", // -ereux > -
— "xue3>", // -eux > -
— "rbé3rbè.", // -ébr > -èbr
— "tur2.", // -rut > -r
— "riré4re.", // -érir > -er
— "rir2.", // -rir > -r
— "câ2ca.", // -âc > -ac
— "snu1.", // -uns > -un
— "rtîa4.", // -aîtr > -
— "long2.", // -gnol > -gn
— "vec2.", // -cev > -c
— "ç1c>", // -ç > -c
— "ssilp3.", // -pliss > -pl
— "silp2.", // -plis > -pl
— "tèhc2te.", // -chèt > -chet
— "nèm2ne.", // -mèn > -men
— "llepp1.", // -ppell > -ppel
— "tan2.", // -nat > -n
— "rvè3rve.", // -èvr > -evr
— "rvé3rve.", // -évr > -evr
— "rè2re.", // -èr > -er
— "ré2re.", // -ér > -er
— "tè2te.", // -èt > -et
— "té2te.", // -ét > -et
— "epp1.", // -ppe > -pp
— "eya2i.", // -aye > -ai
— "ya1i.", // -ay > -ai
— "yo1i.", // -oy > -oi
— "esu1.", // -use > -us
— "ugi1.", // -igu > -g
— "tt1.", // -tt > -t

Références

Alexandre Allauzen and Hélene Bonneau-Maynard. Training and evaluation of pos taggers on the french multitag corpus. In *LREC*, 2008.

Nicolas Béchet. *Extraction et regroupement de descripteurs morpho-syntaxiques pour des processus*

de Fouille de Textes. PhD thesis, Université Montpellier II-Sciences et Techniques du Languedoc, 2009.

Farah Benamara, Carmine Cesarano, Antonio Picariello, Diego Reforgiato Recupero, and Venkatramana S Subrahmanian. Sentiment analysis : Adjectives and adverbs are better than adjectives alone. In *ICWSM*, 2007.

Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka. Measuring semantic similarity between words using web search engines. *www*, 7 :757–766, 2007.

Didier Bourigault. *Lexter : un Logiciel d'EXtraction de TERminologie : application à l'acquisition des connaissances à partir de textes*. PhD thesis, EHESS, 1994.

Didier Bourigault and Cécile Fabre. Approche linguistique pour lâanalyse syntaxique de corpus. *Cahiers de grammaire*, 25 :131–151, 2000.

Didier Bourigault and Christian Jacquemin. Construction de ressources terminologiques. *Ingénierie des langues*, pages 215–233, 2000.

Eric Brill. Transformation-based error-driven learning and natural language processing : A case study in part-of-speech tagging. *Computational linguistics*, 21(4) :543–565, 1995.

André Clas. Collocations et langues de spécialité. *Meta*, 39(4) :576–580, 1994.

B Daille et al. Acabit : une maquette dâaide à la construction automatique de banques terminologiques monolingues ou bilingues. *Class, A., Thoiron, P„ Béjoint (eds) Lexicomatique et Dictionnairiques*, pages 123–136, 1994.

Sophie DAVID and Pierre Plante. Termino version 1.0. *Report, Centre dâAnalyse de Textes par Ordinateur, Université du Québec*, 1990.

Mamadou Dieye, Mohamed Rafik Doulache, Mustapha Floussi, Julie Chabalier, Isabelle Mougenot, Mathieu Roche, et al. Construction d'un dictionnaire multilingue de biodiversité à partir de dires d'experts. In *INFORSID*, 2012.

Ted Dunning. *Statistical identification of language*. Computing Research Laboratory, New Mexico State University, 1994.

Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, volume 7, pages 1606–1611, 2007.

Natalia Grabar and Pierre Zweigenbaum. Acquisition automatique de connaissances morphologiques sur le vocabulaire médical. *Actes de TALN*, 1999 :175–184, 1999.

Gregory Grefenstette. Comparing two language identification schem es. 1995.

Geoff Kuenning, P Willisson, W Buehring, and K Stevens. International ispell. *Version*, 3(00) : 1–33, 2004.

E. Laporte. Mots et niveau lexical. *Ingénierie des langues*, pages 25–49, 2000.

Vladimir I Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. In *Soviet physics doklady*, volume 10, page 707, 1966.

Dekang Lin. An information-theoretic definition of similarity. In *ICML*, volume 98, pages 296–304, 1998.

Donald AB Lindberg and Harold M Schoolman. The national library of medicine and medical informatics. *Western Journal of Medicine*, 145(6) :786, 1986.

Carolyn E Lipscomb. Medical subject headings (mesh). *Bulletin of the Medical Library Association*, 88(3) :265, 2000.

Julie B Lovins. *Development of a stemming algorithm*. MIT Information Processing Group, Electronic Systems Laboratory, 1968.

Alexander Maedche and Steffen Staab. Measuring similarity between ontologies. In *Knowledge engineering and knowledge management : Ontologies and the semantic web*, pages 251–263. Springer, 2002.

AT Mc Cray. A. the unified medical language system. *Meth Inf Med*, 34 :281–291, 1993.

David Milne. Computing semantic relatedness using wikipedia link structure. In *Proceedings of the new zealand computer science research student conference*. Citeseer, 2007.

Fiammetta Namer. Automatiser l'analyse morpho-sémantique non affixale : le système dérif. *Cahiers de grammaire*, 28 :31–48, 2003.

Gonzalo Navarro and Ricardo Baeza-Yates. Very fast and simple approximate string matching. *Information Processing Letters*, 72(1) :65–70, 1999.

Naoaki Okazaki and Sophia Ananiadou. A term recognition approach to acronym recognition. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 643–650. Association for Computational Linguistics, 2006.

Marjorie Paternostre, Pascal Francq, J Lamoral, D Wartel, and M Saerens. Carry, un algorithme de désuffixation pour le français. *Rapport technique du projet Galilei*, 2002.

Ronan Pichon and Pascale Sébillot. Différencier les sens des mots à lâaide du thème et du contexte de leurs occurrences : une expérience. In *6e conférence francophone internationale sur le Traitement Automatique des Langues Naturelles (TALN 99)*, 1999.

Martin F Porter. An algorithm for suffix stripping. *Program : electronic library and information systems*, 14(3) :130–137, 1980.

Fletcher Secret Pratt. Urgent, the story of codes and ciphers blue ribbon books. *Garden City, New York*, 1939.

Thomas S Robertson and Hubert Gatignon. Technology development mode : a transaction cost conceptualization. *Strategic Management Journal*, 19(6) :515–531, 1998.

Mathieu Roche. *Intégration de la construction de la terminologie de domaines spécialisés dans un processus global de fouille de textes*. PhD thesis, Paris 11, 2004.

Mathieu Roche and Violaine Prince. Acrodef : A quality measure for discriminating expansions of ambiguous acronyms. In *Modeling and Using Context*, pages 411–424. Springer, 2007.

Mathieu Roche, Oana Mihaela Garbasevschi, et al. Wemit : Web-mining for translation. In *Conference on Prestigious Applications of Intelligent Systems*, pages 993–994, 2012.

Mehran Sahami and Timothy D Heilman. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on World Wide Web*, pages 377–386. ACM, 2006.

Gerard Salton and Chung-Shu Yang. On the specification of term values in automatic indexing. *Journal of documentation*, 29(4) :351–372, 1973.

Jacques Savoy. A stemming procedure and stopword list for general french corpora. *JASIS*, 50(10) : 944–952, 1999.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of international conference on new methods in language processing*, volume 12, pages 44–49. Manchester, UK, 1994.

Claude E Shannon. A note on the concept of entropy. *Bell System Tech. J*, 27 :379–423, 1948.

Michel Simard. Automatic insertion of accents in french text. In *EMNLP*, pages 27–35. Citeseer, 1998.

Robert L Solso, Paul F Barbuto, and Connie L Juel. Bigram and trigram frequencies and versatilities in the english language. *Behavior Research Methods & Instrumentation*, 11(5) :475–484, 1979.

Sara Tonelli, Elena Cabrio, and Emanuele Pianta. Key-concept extraction from french articles with kx. *Actes de lâatelier de clôture du huitième défi fouille de texte (DEFT)*, pages 19–28, 2012.

Juan Antonio Lossio Ventura, Clement Jonquet, Mathieu Roche, Maguelonne Teisseire, et al. Combining c-value and keyword extraction methods for biomedical terms extraction. In *LBM'2013 : International Symposium on Languages in Biology and Medicine*, pages 45–49, 2013.

George Kingsley Zipf. National unity and disunity. 1941.

Pierre Zweigenbaum. Encoder lâinformation médicale : des terminologies aux systèmes de représentation des connaissances. *Innovation Stratégique en Information de Santé*, 2 :5, 1999.

Pierre Zweigenbaum and Natalia Grabar. Accentuation de mots inconnus : application au thesaurus biomédical mesh. *Proceedings of TALN (Traitement automatique des langues naturelles), ATALA, ATLIF, Nancy*, page 53Á, 2002.