Les supports de vocabulaires pour les systèmes de recherche d'information orientés précision : application aux graphes pour la recherche d'information médicale.

Loïc Maisonnasse

Université Joseph Fourier – Grenoble I

Collège des écoles doctorales

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE JOSEPH FOURIER – GRENOBLE I

Discipline : Informatique

Présentée et soutenue publiquement le 06 mai 2008 par

Loïc MAISONNASSE

TITRE

Les supports de vocabulaires pour les systèmes de recherche d'information orientés précision : application aux graphes pour la recherche d'information médicale

Directeurs de thèse : Mme Catherine BERRUT et M. Jean-Pierre CHEVALLET

Composition du jury

Composition du jury :	
Président :	*Jean-Pierre Giraudin, Professeur, LIG, UPMF, Grenoble*
Rapporteurs :	*Patrick Bosc, Professeur, IRISA, ENSSAT, Lannion*
	Jean-Marie Pinon, Professeur INSA, LIRIS, Lyon
Examinateurs :	*Jean-Pierre Chevallet, Maître de conférence, LIG, UPMF, Grenoble*
	Catherine Berrut, Professeur, LIG, UJF, Grenoble

Thèse préparée au sein de l'équipe MRIM du laboratoire LIG

(Laboratoire Informatique de Grenoble)

Université Joseph Fourier – Grenoble I

REMERCIEMENTS

Je tiens à remercier,

Monsieur Jean-Pierre Giraudin, Professeur à l'Université Pierre Mendès France, qui m'a fait l'honneur de présider ce jury ;

Monsieur Jean-Marie Pinon, Professeur à l'Institut National des Sciences Appliquées (INSA) de Lyon, qui a bien voulu rapporter ce travail, ainsi que pour l'intérêt qu'il a porté à l'égard de ce dernier.

Monsieur Patrick Bosc, Professeur à l'École Nationale Supérieure des Sciences Appliquées et de Technologie (ENSSAT), qui a bien voulu rapporter ce travail, et qui par ces critiques constructives m'a permis d'améliorer la qualité de ce rapport ;

Madame Catherine Berrut, Professeur à l'Université Joseph Fourier et responsable de l'équipe de recherche MRIM, qui m'a accueilli dans son équipe et qui a dirigé mes travaux de recherche. Et qui par ces remarques constructives m'a permis de construire une thèse cohérente ;

Monsieur Jean-Pierre Chevallet, Maître de Conférence à l'université Pierre Mendès France, qui a co-dirigé mes travaux de recherche malgré la distance. Il a su, par ces remarques et ces conseils, me guider à travers les méandres de la recherche ;

Monsieur Eric Gaussier, Professeur à l'Université Joseph Fourier, sans qui toute une partie de cette thèse ne serait pas complète ;

Les membres actuels ou anciens de l'équipe MRIM pour leur accueil dans l'équipe, leur soutien et leurs conseils qui ont fait avancer ce travail, ainsi que les membres du laboratoire LIG ;

Les collègues du labo, Ali, Caroline, Delphine, Didier, Helga, Leila, Rami, Stéphane, Trong Ton, qui m'ont supporté dans leur bureau ou aux pauses cafés. Et sans qui les journées de travail n'auraient pas été aussi agréables ;

Mes amis, dont je ne citerai pas les noms, qui m'ont fourni tout au long de ces années les moments de détente nécessaires au maintien de mon état mental. Ceux qui ont participé aux soirées vin, jeu de rôle, barbecues ou aux apéros en tous genres. J'espère qu'ils se reconnaîtront dans ces quelques lignes ;

Ma famille qui a toujours été présente, et qui m'a toujours soutenu même si mon travail leur a toujours semblé ésotérique ;

Mes parents sans qui rien n'aurait été possible, et qui m'ont accompagné tout au long de mes études ;

Carine qui m'a soutenu pendant toutes ces années, qui a subi mes humeurs pour le moins changeantes, et qui a toujours cru en moi. Pour cela, je lui suis extrêmement reconnaissant.

TABLE DES MATIERES

TABLES DES ILLUSTRATIONS

TABLE DES TABLEAUX

PARTIE 1 : INTRODUCTION

Chapitre I Contexte

« Sous l'avalanche ininterrompue d'informations insignifiantes, plus personne ne sait où puiser les informations intéressantes. » Bernard Werber (La révolution des fourmis)

Nous évoluons aujourd'hui dans une société de l'information. Dans ce contexte, le volume d'informations, notamment textuelle, échangé et stocké à travers le monde devient de plus en plus grand. L'accès à cette information constitue un enjeu de société, aussi bien pour les entreprises, qui doivent se tenir à jour des dernières innovations et être réactives face à des informations internes ou externes, que pour les particuliers, qui par l'intermédiaire d'Internet demandent de plus en plus d'information.

Permettre aux personnes en quête d'information de localiser rapidement et efficacement une information dans un vaste ensemble de documents électroniques est essentiel.

1 La recherche d'information

La recherche d'information se définit comme un domaine de recherche en informatique qui s'attache à définir des modèles et des systèmes, dans le but de faciliter l'accès à l'information contenue dans des documents.

Initialement réduite à la recherche de documents tels que des versions électroniques de documents papiers (Salton, 1971), la notion de document a beaucoup évolué. Cette notion s'applique maintenant à tous les types de médias : texte, image, son ou vidéo, ainsi qu'aux documents structurés ou composés de plusieurs de ces médias.

Les systèmes de recherche d'information ne répondent pas directement à la question d'un utilisateur. Ces systèmes retournent à l'utilisateur un ensemble de documents sensés contenir des informations qui répondent à son besoin. L'extraction à partir des documents de ces informations reste à la charge de l'utilisateur.

On peut caractériser un système de recherche d'information comme un outil informatique capable de retrouver les documents pertinents pour le besoin d'information d'un utilisateur exprimé sous la forme d'une requête.

2 La nécessité d'une recherche d'information orientée précision

Avec l'augmentation du volume d'informations électroniques échangé et stocké, la recherche d'information fait face à de nouveaux défis. En effet, avec l'accroissement du nombre de documents, la similitude entre ces derniers augmente. Pour un besoin d'information, le nombre de documents plus ou moins satisfaisants devient grand. Les utilisateurs ne peuvent pas toujours explorer l'ensemble de ces documents, notamment dans des contextes professionnels du fait de contraintes de temps et d'efficacité.

Dans des contextes professionnels tels que la médecine, ou l'archéologie, les utilisateurs expriment des besoins d'information experts, c'est-à-dire des besoins d'information détaillés. Dans ces contextes, les utilisateurs élaborent des requêtes complexes qui représentent ces besoins d'information experts, comme l'illustre la requête *'Carcinome à cellules de Merkel'* pour le domaine médical. Satisfaire au maximum les utilisateurs nécessite alors de retrouver des documents qui répondent complètement et précisément à leurs besoins d'information. Cela peut nécessiter de trouver des réponses dans plusieurs langues, par exemple des documents contenant *'Merkel cell carcinoma'*. Les systèmes de recherche d'information se doivent donc d'être particulièrement performants au niveau des premières réponses fournies à l'utilisateur, d'autant plus si cet utilisateur exprime des besoins experts.

Les systèmes actuels ne prennent pas en compte les besoins d'information experts, la plupart se limitent à l'utilisation de mots-clefs. Même si ces systèmes proposent une formulation de requêtes complexes sous forme de phrases, ils découpent cette phrase en mots-clefs, et ils ne prennent pas en compte son caractère complexe et structuré. Par conséquent, les réponses proposées pour de telles requêtes sont partielles et imprécises et elles ne sont données que dans une seule langue, celle de la requête.

Répondre à des besoins d'information experts nécessite l'utilisation de systèmes de *recherche d'information orientés précision*. Un système orienté précision doit fournir les documents les plus pertinents pour un besoin d'information expert dès les premiers documents retrouvés par le système et si possible dans l'ensemble des langues compréhensibles par l'utilisateur. Cela peut impliquer d'éliminer certains documents parmi des documents équivalents, au risque de diminuer le rappel. Pour atteindre cette précision, un système doit utiliser des représentations expressives des documents et des requêtes de façon à prendre en compte leurs différents aspects. L'expressivité d'une représentation passe par la multiplication des points de vue sur ce qu'elle représente, et par la couverture de l'information véhiculée par ces points de vue.

Dans une certaine mesure, la recherche d'information orientée précision se situe dans un intervalle entre la recherche d'information habituelle et les systèmes de question-réponse ; en effet un système de question-réponse propose de répondre à des besoins très précis exprimés sous forme de questions. Répondre correctement à ces questions nécessite des traitements linguistiques particuliers des questions et des documents, généralement plus fins que ceux utilisés en recherche d'information. Un système d'information orienté précision doit lui aussi analyser en profondeur le contenu des documents et le contenu des requêtes pour en donner des représentations expressives. Ces représentations permettent ensuite de déterminer les documents les plus pertinents.

L'intérêt de l'expressivité se retrouve dans le scénario décrit dans le rapport du projet PRISM (Sérasset, 2003) : un utilisateur veut effectuer une étude sur *'l'éducation par la recherche'* par l'intermédiaire d'Internet. Pour prospecter les documents qui l'intéressent, cet utilisateur accède à un moteur de recherche quelconque et saisit la requête '*l'éducation par la recherche'*. Le résultat retourné par les moteurs de recherche actuels fournit de nombreux documents en français concernant tout un ensemble de domaines englobants mais pas forcément pertinents : *l'éducation par la recherche, l'éducation à la recherche, la recherche au ministère de l'éducation, la recherche en éducation, le moteur de recherche de l'institut d'éducation à la santé, etc.* Nous prenons ici (figure 1) l'exemple des résultats à cette requête fournis par Google. Utiliser une expressivité plus forte, notamment en exprimant les relations de la phrase, permettrait de sélectionner les bonnes réponses. De plus, une représentation expressive interlingue qui représenterait différentes langues par formalisme unique permettrait de retrouver des documents dans plusieurs langues.

L'utilisation de représentations plus expressives obtenues par une analyse plus précise et une meilleure compréhension des documents et des requêtes, nécessite l'utilisation d'informations

complémentaires. Ces informations complémentaires sont de natures linguistiques, statistiques ou encore sémantiques.


Ministère de l'**Education** nationale, de la **Recherche** et de la **...**
Site officiel. Présentation du système éducatif, les types d'enseignement, la formation continue et les liens internationaux. Formulaires administratifs et **...**
www.education.gouv.fr/ - 34k - En cache - Pages similaires

Institut national de **recherche** pédagogique (INRP)
Départements de **recherche** : Mémoire de l'**éducation**, Ressources et communication, Didactiques des disciplines, Technologies nouvelles et **éducation**, **...**
www.inrp.fr/ - 31k - En cache - Pages similaires

Recherche en **éducation** : actualités et ressources.
Dernières mises à jour · **Recherche** par thèmes **...** Les cahiers d'**Éducation** et devenir - N°7 mai 2006- Le handicap à l'école, Lire... **...**
www.inrp.fr/vst/ - 17k - 20 août 2006 - En cache - Pages similaires
[Autres résultats, domaine www.inrp.fr]

CGT **Education** culture
Fédération de l'**éducation**, de la **recherche** et de la culture (Ferc-CGT). Activité fédérale et par branche. Adresses.
www.ferc.cgt.fr/ - 2k - En cache - Pages similaires

Figure 1 Résultats de Google à la requête 'l'éducation par la recherche' le 22/08/2006

3 L'expressivité en recherche d'information

La plupart des SRI actuels sont basés sur des distributions statistiques de mots-clefs et semblent atteindre une limite. Cette limite est plus visible dans le cas d'un SRI orienté précision comme on l'a vu dans l'exemple précédent. Nous faisons alors l'hypothèse que cette limite ne peut se franchir que par l'utilisation de représentations expressives.

Pour construire ces représentations expressives, il faut disposer de structures statistiques, syntaxiques, ou sémantiques. Ces structures peuvent être de natures :

- endogènes, c'est-à-dire extraites du corpus par un calcul de collocation, ou alors par des analyses syntaxiques ou morphologiques ;
- exogènes, c'est-à-dire provenant de ressources extérieures produites manuellement, structurées (ex: thésaurus, ontologie), ou non (ex: wikipédia) ou encore des ressources produites automatiquement à l'aide de corpus d'apprentissage.

Sur le texte, de nombreux outils de traitement de la langue permettent de construire des représentations expressives des documents. Ces outils se basent sur des approches théoriques qui mettent en avant des degrés d'expressivité et qui proposent des représentations pour ces différents degrés. Nous devons déterminer les outils pertinents pour la recherche d'information orientée précision. Nous nous questionnons donc sur l'intégration de ces informations au sein des modèles de recherche d'information et sur la modélisation de l'expressivité qu'elles apportent.

4 L'expressivité fondée sur l'information linguistique

Certains travaux de recherche d'information se basent sur la théorie Sens Texte, notamment pour intégrer l'utilisation de paraphrases (Wallis, 1993). Cette théorie présente différents niveaux d'expression du sens. Pour introduire la notion d'information linguistique, nous présentons donc la théorie linguistique Sens Texte, avant de détailler ses particularités par rapport à l'expressivité en recherche d'information.

4.1 La théorie Sens Texte

La théorie Sens Texte a été développée par Mel'čuk (Mel'čuk, 1997). Cette théorie sémantique propose un cadre pour l'étude des langues naturelles basé sur la construction de modèles des langues. L'étude de la sémantique ne se définit pas simplement comme l'étude du sens mais comme l'étude de la relation entre les sens et l'expression écrite ou orale de ces sens. L'essence du sens forme une représentation dans l'esprit des personnes ; par nature le sens reste difficile à modéliser. La théorie Sens Texte postule la création d'un ensemble de représentations sémantiques isomorphes aux représentations des sens d'un utilisateur dans une langue. Elle considère la langue comme un mécanisme ayant deux fonctions[1] :

- Parler : définit la méthode qui permet de transformer une représentation du sens en un texte. Cela consiste à faire correspondre à un sens tous les textes d'une langue qui véhiculent ce sens et choisir celui dont l'expression est adaptée aux circonstances.
- Comprendre la parole : définit, à l'inverse, la méthode qui reconstruit une telle représentation à partir des mots d'un discours. Cela revient à faire correspondre à un texte tous les sens possibles pour ce texte et sélectionner le sens pertinent pour le contexte.

L'auteur définit une relation n-aire entre textes et sens, décrivant ainsi les ambiguïtés de la langue et la nécessité de prendre en compte le contexte pour résoudre ces ambiguïtés (figure 2).

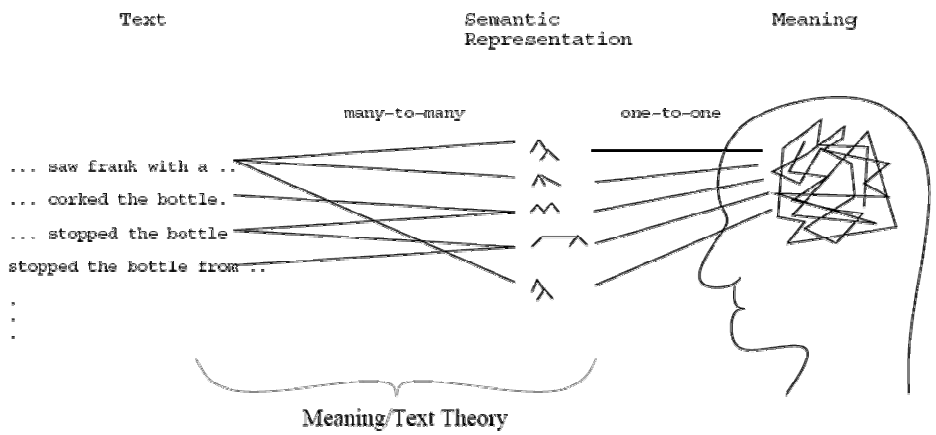


*Figure 2 Lien entre les sens et leurs expressions (*Wallis, 1993*)*

[1] Ces deux fonctions contiennent les fonctions Écrire et Lire dont les traitements se termine au niveau morphologique.

La théorie Sens Texte se base sur trois postulats (Kahane, 2001) :

- La langue désigne un ensemble de règles qui permet de faire correspondre un ensemble de sens et un ensemble de textes. La modélisation d'une langue correspond à la modélisation de la correspondance entre l'ensemble des sens de cette langue et l'ensemble ses textes.
- Une correspondance Sens Texte se décrit par un système formel simulant l'activité linguistique d'un sujet. La modélisation d'une langue spécifie le processus par lequel le locuteur transforme ce qu'il veut dire (Sens) en ce qu'il dit (Texte).
- La correspondance prend une forme modulaire et contient des niveaux de représentation intermédiaires entre le sens et le texte. La théorie Sens Texte postule deux niveaux intermédiaires entre le niveau sémantique (sens) et le niveau phonétique (Texte) : le niveau syntaxique et le niveau morphologique. Chacun de ces niveaux se subdivise en un sous-niveau profond et un sous-niveau de surface. La correspondance Sens Texte s'effectue par l'enchaînement de modules réalisant la correspondance entre les différents niveaux intermédiaires. La figure 3 résume ces niveaux.

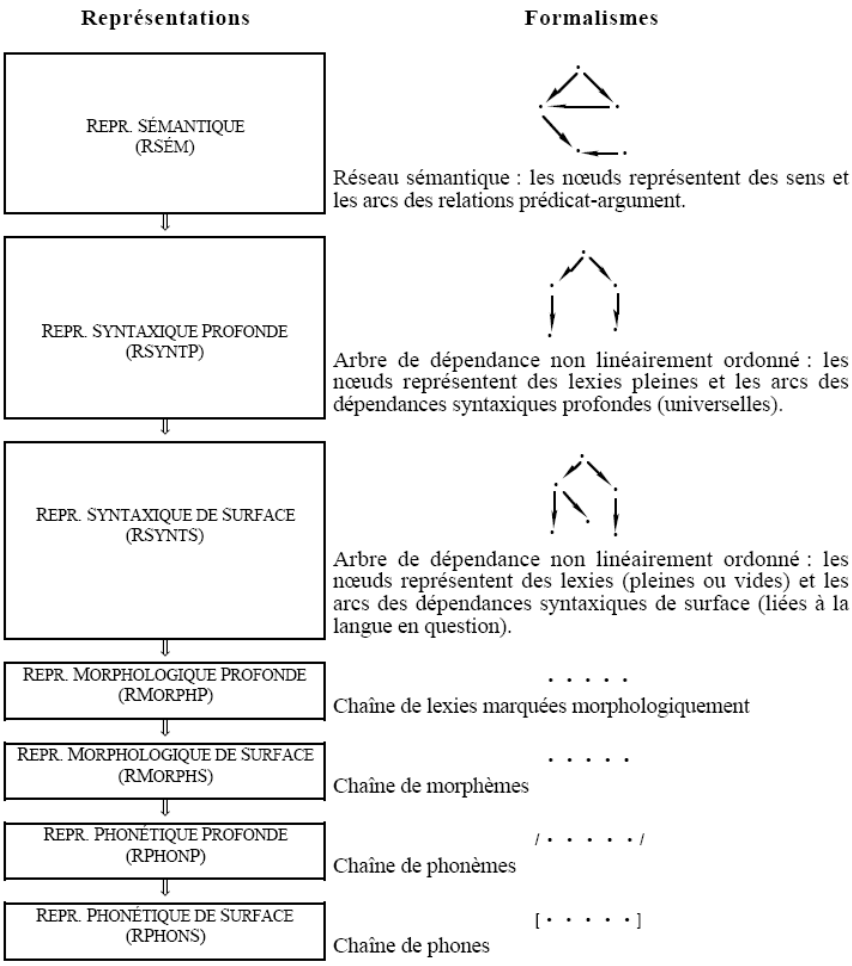


*Figure 3 Niveaux de la théorie Sens Texte (*Polguère, 1998*)*

Une caractéristique intéressante qui découle de ces postulats provient du fait que les modèles Sens Texte se basent sur les paraphrases. La capacité première du locuteur consiste à produire pour un sens

de départ l'ensemble des textes qui peuvent exprimer ce sens, et ensuite à choisir celui qui correspond le mieux au contexte.

4.2 La recherche d'information vue par la théorie Sens Texte

La tâche de recherche d'information peut se percevoir à travers le filtre de la théorie Sens Texte. Un utilisateur cherche à résoudre un besoin d'information, pour cela il se représente un ou plusieurs sens pertinents pour son besoin d'information. Pour chacun de ces sens, l'utilisateur produit l'ensemble des textes qui peuvent les exprimer. Ensuite, dans un système basé sur les mots-clefs, l'utilisateur sélectionne dans ces textes un ensemble de mots représentatifs de son besoin d'information. Dans un système où l'utilisateur fournit une phrase en requête, l'utilisateur sélectionne la phrase représentative de son besoin d'information. Pour un utilisateur, la sélection de documents pertinents consiste donc à rechercher des documents où il retrouve des sens qui correspondent aux sens exprimant son besoin d'information. Dans le cadre de la théorie Sens Texte, la fonction suivante peut être définie :

- Résoudre un besoin d'information : la méthode qui consiste à rechercher dans des textes tous les sens qui correspondent aux sens représentant le besoin d'information.

Ainsi, un système orienté précision capable de représenter les documents et les requêtes à l'aide d'une représentation sémantique permet de résoudre plus précisément les besoins d'information.

4.3 Conclusion

Un sens se représente par une variété de représentations dans une langue, il s'agit d'une représentation expressive. Avoir une description au niveau sens d'un besoin d'information permet de couvrir plusieurs des instanciations possibles de ce sens au niveau textuel. Une représentation s'éloignant du niveau phonétique et s'approchant d'un niveau sémantique améliore les performances de recherche d'information. En effet, une telle représentation des documents prend mieux en compte les variations et les paraphrases, et par conséquent répond mieux à des besoins précis.

5 Bilan

Définir un système orienté précision nécessite d'utiliser des systèmes qui représentent le contenu des documents et des requêtes de façon expressive. L'expressivité d'une représentation se détermine par l'ensemble de ce qu'elle peut exprimer. Ces représentations expressives se construisent à l'aide d'informations diverses permettant de créer un ou plusieurs points de vue sur le document.

Si la théorie sens texte ne peut pas s'appliquer directement en recherche d'information, car elle est trop liée à la modélisation de la langue, les notions qu'elle met en avant peuvent être reprises pour inspirer la construction des représentations des documents.

Nous proposons de définir des modèles orientés précision, c'est-à-dire utilisant un ou plusieurs descripteurs, si possible sémantiques, capables de résoudre des requêtes expertes. Pour cela nous proposons un cadre de définition des modèles de recherche d'information qui met en avant l'expressivité.

Chapitre II Positionnement de la Thèse

"A clever person solves a problem. A wise person avoids it." Einstein

Peu de travaux énoncent clairement l'expressivité des systèmes de recherche d'information. Or modéliser clairement cette expressivité permet de comparer et de positionner ces systèmes les uns par rapport aux autres. Nous proposons, dans cette thèse, d'étudier et de modéliser l'expressivité notamment pour la recherche d'information orientée précision. Pour cela nous suivons les étapes décrites sur la figure 4. Nous présentons dans une première partie notre approche de la modélisation de l'expressivité. Pour cela nous caractérisons cette expressivité et son intérêt en recherche d'information. Nous proposons ensuite une modélisation de cette expressivité qui passe par la définition d'un cadre de modélisation qui utilise des *supports de vocabulaires*. À l'aide de ce cadre nous proposons dans une deuxième partie deux modèles qui explorent les propriétés offertes par notre modélisation de l'expressivité : le modèle local et le modèle global. Nous détaillons enfin dans la dernière partie nos choix sur l'application de ces modèles à la recherche d'information orientée précision. Nous détaillons le niveau d'expressivité choisi, le type de représentation et les méthodes permettant leur obtention.

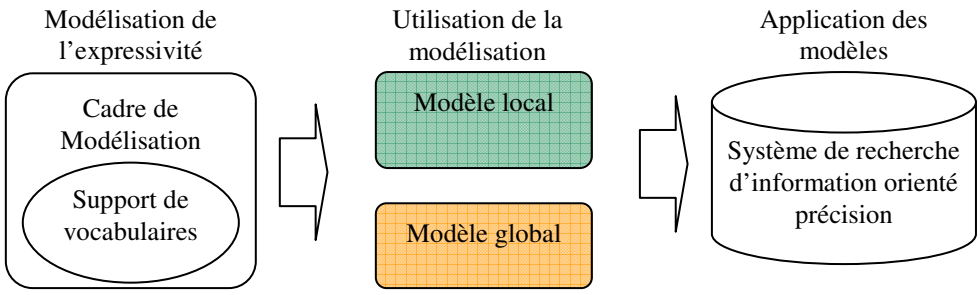


Figure 4 Enchaînement des choix effectués pour l'étude de l'expressivité

1 Modélisation de l'expressivité par les supports de vocabulaires

1.1 Présentation de l'expressivité

L'expressivité représente le nombre de points de vue utilisés pour représenter les documents et l'espace d'expression de chacun de ces points de vue. Sur un même document, plusieurs points de vue peuvent être proposés. Par exemple, sur la figure 5, nous proposons plusieurs points de vue sur une même image ; l'un à l'aide d'un histogramme de couleur, l'autre à l'aide de mots désignant ce qui apparaît sur l'image, ou encore un dernier à l'aide de mots donnant un sens à l'image. Ces points de vue n'ont pas tous le même espace d'expression, par contre ils peuvent être complémentaires et peuvent être utilisés de manière conjointe.

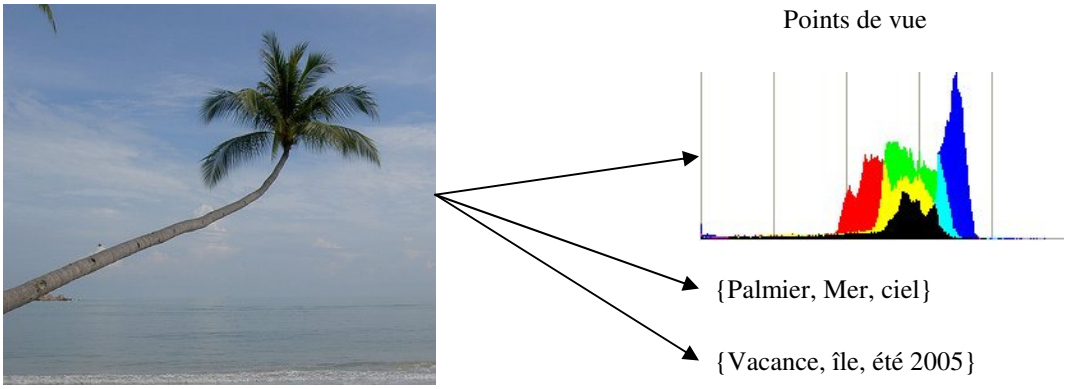


Figure 5 Expression de différents points de vue sur une image

Dans cette thèse nous nous intéressons essentiellement à l'information textuelle. Pour cette raison, nous posons la simplification que l'expressivité peut se représenter sur un axe dérivé de celui proposé par la théorie Sens Texte, tel que présenté sur la figure 6. Cet axe relie à ses extrémités les systèmes à expressivité faible et ceux à expressivité forte. Les systèmes à expressivité faible représentent le document à l'aide d'un seul point de vue et ce point de vue utilise des descripteurs peu expressifs. C'est le cas des systèmes à base de mots-clefs qui constituent la majorité des systèmes de recherche d'information actuels. Les systèmes à expressivité forte représentent les documents à l'aide de plusieurs points de vue et utilisent des descripteurs expressifs tels que des descripteurs sémantiques. Par exemple, les systèmes à base d'expressions complexes sont expressifs, leurs représentations fournissent plusieurs points de vue sur le document et ces points de vue sont de niveau sémantique pour les plus expressifs.

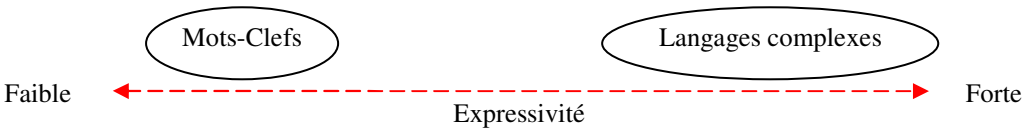


Figure 6 Axe de l'expressivité des systèmes de recherche d'information

L'expressivité d'un système de recherche d'information se manifeste à travers ses représentations des documents et des requêtes ; en premier par le nombre de points de vue adoptés pour représenter le document, ce qui correspond aux différents types de descripteurs utilisés ; elle se manifeste ensuite par l'expressivité de ses descripteurs.

Modéliser l'expressivité permet de sélectionner les langages d'indexation dont l'expressivité convient le mieux à une tâche donnée. Actuellement, il n'existe pas de modélisation générique qui met en avant l'expressivité, et de fait, comparer ou positionner des systèmes sur leur niveau d'expressivité se révèle difficile. Pourtant une telle modélisation est particulièrement intéressante pour la recherche d'information orientée précision où l'expressivité constitue un point majeur du système.

Afin de répondre à ce besoin, nous proposons un cadre pour la définition de modèles de recherche d'information qui modélise l'expressivité. Nous proposons et nous définissons ainsi la notion de support de vocabulaires telle que décrite dans la figure 7.

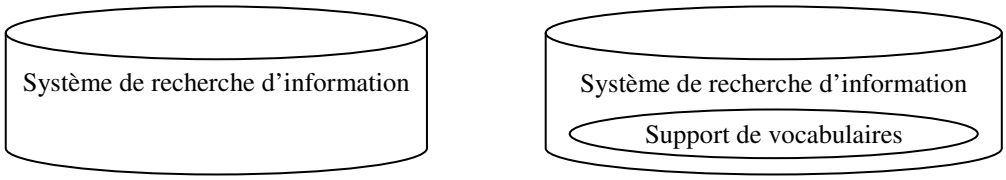


Figure 7 Ajout du support de vocabulaires dans le système de recherche d'information

1.2 Le support de vocabulaires pour modéliser l'expressivité

Nous définissons des modèles dans lesquels des *supports de vocabulaires* modélisent l'expressivité. Le support de vocabulaires définit l'ensemble des *vocabulaires* utilisés pour représenter les documents ou les requêtes. Un vocabulaire correspond à un point de vue sur le document, il est constitué d'un ensemble de descripteurs, ou *unités de vocabulaires*.

La majorité des modèles de recherche d'information possèdent une expressivité faible, ils n'utilisent qu'un seul vocabulaire formé d'une seule sorte de descripteurs. Ce vocabulaire se compose de descripteurs simples éventuellement associés à un poids. Par exemple, dans les modèles vectoriels sur les mots-clefs, le support de vocabulaires est constitué d'un seul vocabulaire : l'ensemble des mots-clefs pondérés.

Sur d'autres médias, des modèles utilisent plusieurs vocabulaires, cela permet d'exprimer différents points de vue. Par exemple, une vidéo peut se représenter à l'aide d'un vocabulaire visuel et d'un vocabulaire auditif. Dans ce cas, le support de vocabulaires contient les deux vocabulaires qui fournissent deux points de vue sur le document.

Le support de vocabulaires dénote l'expressivité de la représentation par deux aspects :

- Le premier aspect correspond au nombre de vocabulaires utilisés. Cela constitue le nombre de points de vue par lesquels le système interprète le document ou la requête.
- Le deuxième aspect correspond à l'expressivité de chaque vocabulaire utilisé. Cela dépend de la définition des unités de vocabulaires. Les mots-clefs constituent des unités de vocabulaires simples dont le niveau d'expressivité correspond au niveau morphologique. Les relations syntaxiques constituent des unités de vocabulaires complexes formées de plusieurs éléments : deux lemmes et une étiquette de relation. Leur niveau d'expressivité correspond au niveau syntaxique.

Ces deux aspects nous permettent de positionner les systèmes les uns par rapport aux autres sur l'axe de l'expressivité de la figure 6.

Les supports de vocabulaires, tels que nous les proposons, permettent de modéliser l'expressivité. Ils modélisent le langage d'indexation (langage de représentation des documents) et le langage d'interrogation (langage de représentation des requêtes). Nous les utilisons pour définir un cadre dans lequel différents modèles peuvent s'exprimer. Par conséquent, ce cadre permet de comparer plus facilement des systèmes entre eux.

1.3 Utilisation du support de vocabulaires

Définir les supports de vocabulaires fixe l'expressivité d'un système de recherche d'information. Indépendamment de l'expressivité, les systèmes basés sur des supports de vocabulaires se distinguent

par la manière dont ils utilisent leurs supports de vocabulaires. Nous nous focalisons sur l'utilisation des supports de vocabulaires lors de l'indexation et nous considérons que tous les vocabulaires du support possèdent un comportement identique.

Nous présentons deux aspects portant sur la manipulation du support de vocabulaires lors de l'indexation des documents. Ces aspects permettent de distinguer différents modèles d'indexation sur des critères indépendants de l'expressivité.

Le premier aspect correspond à la *portée des vocabulaires*. Cette portée représente le nombre d'unités de vocabulaires disponibles pour représenter les documents. De nombreux systèmes limitent le vocabulaire utilisable. C'est le cas du modèle vectoriel qui se limite soit à une liste d'autorité, soit aux mots de la collection. Cette limite est plus ou moins contraignante, elle se représente par l'axe *portée des vocabulaires* de la figure 8. Sur cet axe les vocabulaires peuvent être soit exhaustifs soit spécifiques :

- Un *vocabulaire exhaustif* définit un ensemble complet d'unités de vocabulaires, par exemple toutes les chaînes de caractères possibles ou tous les concepts définis par une ressource. C'est le cas du modèle logique sur les mots-clefs, où le modèle utilise n'importe quelle chaîne de caractères.
- Un vocabulaire *spécifique* définit un ensemble d'unités de vocabulaires borné par des contraintes, par exemple les mots appartenant au corpus, ou une sélection de concepts en fonction d'une liste d'autorité. C'est le cas du modèle vectoriel sur les mots-clefs, où le modèle n'utilise que les mots-clefs qui forment les dimensions du vecteur. Par exemple, dans le domaine médical, on n'utilisera que des mots-clefs du domaine.

Le deuxième aspect correspond à la *portée de ces représentations*, c'est-à-dire la construction des représentations de document à partir du support de vocabulaires. Cette portée représente la part des unités de vocabulaires utilisée par le modèle de recherche d'information pour représenter un document. La représentation du document peut donc se faire sur une partie limitée des unités de vocabulaires définies par le support de vocabulaires. Par exemple, le modèle vectoriel sur les mots-clefs ne représente un document que par les mots-clefs qui apparaissent dans ce document. La représentation du document peut aussi se faire sur l'ensemble du support de vocabulaires, c'est le cas dans les modèles de langue où le modèle d'un document est calculé sur l'ensemble des mots constituant le support de vocabulaires.

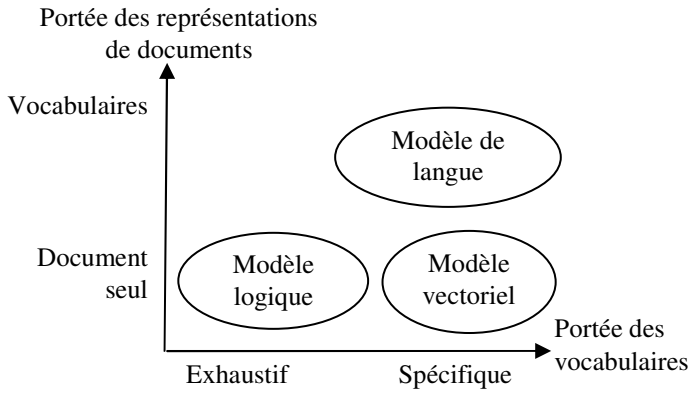


Figure 8 Plan formé par l'utilisation des supports de vocabulaires pour l'indexation, avec le positionnement de trois modèles sur cet espace

La représentation des documents peut utiliser une part plus ou moins grande du support de vocabulaires ; elle n'utilise au minimum que les éléments du document, au maximum l'ensemble des éléments du vocabulaire. Elle peut donc se représenter sous la forme de l'axe *portée des représentations de documents* de la figure 8.

2 Deux utilisations du support de vocabulaires : modèle local et modèle global

Nous utilisons des systèmes orientés précision en nous appuyant sur les supports de vocabulaires pour modéliser l'expressivité. Volontairement, pour analyser au mieux l'apport des supports de vocabulaires, nous proposons des modèles d'expressivité équivalente. En nous abstrayant de l'expressivité, nous différencions ces modèles par leur positionnement sur l'espace d'utilisation des supports de vocabulaires.

Plusieurs positionnements sont possibles dans cet espace, cependant nous ne les abordons pas tous pour deux raisons principales :

- D'une part car la mise en œuvre de certaines positions est complexe. Par exemple représenter un document sur l'ensemble des vocabulaires d'un support exhaustif nécessite de représenter chaque document à l'aide de tous les éléments d'un vocabulaire potentiellement très grand.
- D'autre part car la différence entre certaines positions a déjà été étudiée. Par exemple représenter uniquement le document et faire varier la portée des vocabulaires revient à utiliser ou non un vocabulaire contrôlé.

Nous choisissons donc de présenter deux modèles qui s'opposent dans cet espace et qui sont proches de certains modèles habituellement utilisés en recherche d'information. Ces deux modèles sont fondés sur des supports de vocabulaires et se positionnent à l'opposé sur le plan de l'utilisation du support de vocabulaires (cf. figure 9) :

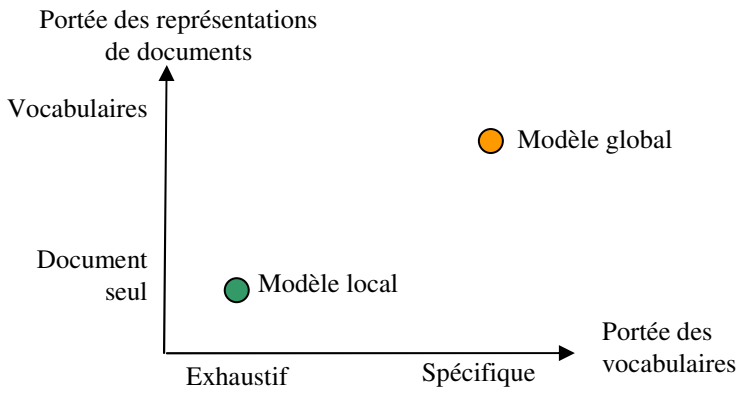


Figure 9 Positionnement des deux modèles dans le plan formé par l'utilisation des supports de vocabulaires pour l'indexation

- Le premier, nommé modèle **local**, utilise des vocabulaires *exhaustifs*. Sur ce support de vocabulaires, le modèle local n'emploie que les unités des vocabulaires détectées **localement** dans les documents. Ce positionnement correspond à celui habituellement utilisé par les modèles logiques.

- Le second, nommé modèle **global**, utilise des vocabulaires *spécifiques* limités à l'utilisation du vocabulaire de la collection. Sur ce support, le modèle global utilise la **globalité** des unités de vocabulaires pour représenter un document. Ce positionnement est par exemple utilisé par les modèles de langue en recherche d'information.

Ces deux modèles manipulent les supports de vocabulaires de différentes manières, par conséquent les modèles sous-jacents de recherche d'information utilisés dépendent de ces contraintes. Dans la suite, nous présentons le type de représentation et le niveau d'expressivité choisis pour représenter les documents. Puis nous caractérisons les choix permettant de créer cette représentation à partir du texte.

3 Application des modèles local et global pour une expressivité forte

Nous souhaitons proposer deux modèles orientés précision. Par conséquent, sur l'axe de l'expressivité, nous positionnons le modèle local et le modèle global à un niveau d'expressivité fort. En combinant l'axe de l'expressivité et le plan d'utilisation des vocabulaires, nous pouvons positionner nos deux modèles tels que le représente la figure 10.

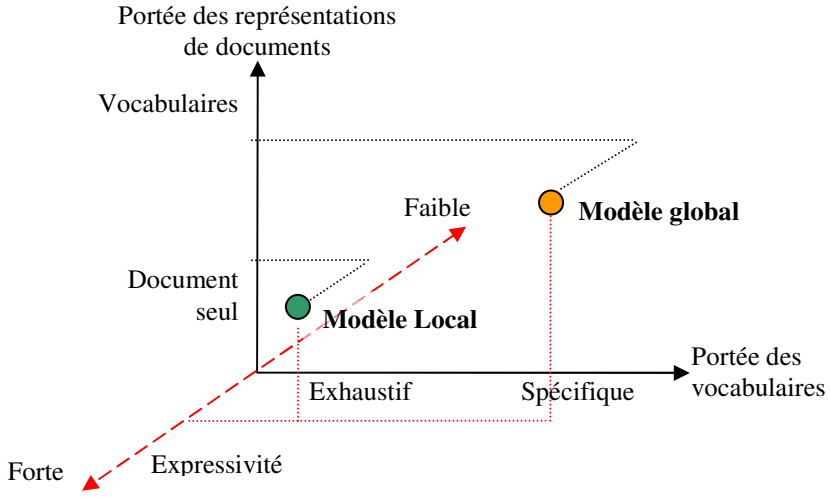


Figure 10 Positionnement des deux modèles dans l'espace formé par le plan de l'utilisation des supports de vocabulaires pour l'indexation combiné à l'axe de l'expressivité

Nous proposons de baser ces deux modèles sur des représentations conceptuelles à base de graphes. Par conséquent leurs supports possèdent une expressivité similaire. Ces supports représentent deux points de vue sémantiques sur le document. Le premier représente la vision conceptuelle du document par l'utilisation d'un vocabulaire à base de concepts, le second représente la vision relationnelle du document par l'utilisation d'un vocabulaire à base de relations sémantiques.

L'intérêt des graphes est de fournir un cadre simple et facilement adaptable. Les modèles qui utilisent cette représentation peuvent se simplifier ou s'adapter à d'autres représentations, telles que des arbres, avec un minimum d'effort. Une modélisation effectuée dans ce cadre peut alors s'adapter à d'autres niveaux d'expressivité, par exemple au niveau syntaxique. En effet, si certains domaines, comme la médecine, possèdent de nombreuses bases de connaissances, cela n'est pas vrai pour tous les domaines où les représentations sémantiques sont impossibles à construire. Un modèle qui s'adapte à différents niveaux d'expressivité est utilisable sur des domaines variés.

3.1 Les représentations conceptuelles pour une expressivité forte

3.1.1 Intérêt des représentations conceptuelles

Nous proposons d'utiliser le traitement de la langue pour produire des représentations à base de graphes. Dans ces représentations, nous utilisons des concepts plutôt que des mots ou des termes. Indexer à l'aide de termes (e.g. '*tomographie axiale calculée par ordinateur'*) améliore la précision car les termes sont moins ambigus que les mots. Cependant cela mène à des problèmes de rappel provenant de la variation des termes et de la synonymie (e.g. '*tomodensitométrie'*). Indexer au niveau conceptuel permet de résoudre ces problèmes car les concepts constituent une abstraction des termes et résolvent les problèmes dus aux variations d'expression.

Les concepts se définissent comme des notions abstraites uniques, compréhensibles par l'homme, indépendantes de n'importe quel support matériel, langue ou représentation. Une indexation conceptuelle représente donc des documents dans différentes langues par un format unique, ce qui aboutit naturellement à une indexation interlingue. Cela améliore les performances du système orienté précision qui retourne des réponses quelle que soit la langue.

Pour des requêtes courtes et expertes, les relations sémantiques entre les concepts fournissent des informations importantes. En effet, elles donnent aux concepts un rôle en les comparant aux autres concepts de la phrase. Un utilisateur a besoin de retrouver la même relation sémantique dans sa requête et dans les documents retrouvés (e.g. '*tomodensitométrie d'un emphysème'*).

3.1.2 Les ressources du domaine

La construction d'une représentation conceptuelle passe par l'utilisation de ressources contenant des informations. Construites par des spécialistes, ces ressources doivent incorporer l'ensemble des termes et leurs variations utiles pour le domaine (terminologie). Chacun de ces termes doit se relier correctement à des concepts. La ressource doit aussi décrire les relations possibles entre ces concepts (e.g. '*localisation de (emphysème, image de tomodensitométrie)*'). Nous notons qu'il est illusoire qu'une source de connaissances contente tous les praticiens d'un domaine de spécialité car toute classification/organisation contraint la réalité dans un point de vue toujours discutable. Ce point important implique que l'usage d'une ressource pour l'indexation revient à forcer un point de vue à tout utilisateur. Nous considérons que ces ressources découlent d'un consensus de spécialistes, ce qui se vérifiera dans les expérimentations ; par conséquent la ressource utilisée représente une conceptualisation communément admise sur le domaine.

Dans cette thèse nous nous intéressons au domaine médical, un domaine d'expertise pour lequel des ressources de bonne qualité existent. Nous choisissons plus précisément la ressource d'information médicale UMLS[2]. UMLS fournit une conceptualisation du domaine médical, elle définit des concepts et des relations sémantiques entre ces concepts.

3.2 Les représentations expressives appliquées au texte

3.2.1 Positionnement des représentations expressives

Contrairement à la théorie Sens Texte qui produit des structures représentant le sens au niveau de la phrase, la recherche d'information a pour objectif de déterminer les blocs d'informations pertinents pour un besoin d'utilisateur. En recherche d'information, la phrase représente un bloc trop petit pour répondre à un besoin d'information, même précis. La recherche d'information, au contraire, cherche à

[2] http://umlsks.nlm.nih.govumlsks.nlm.nih.gov/

fournir des éléments d'information sur la *thématique (aboutness)* décrite par la requête, c'est-à-dire au sujet du besoin de l'utilisateur.

La recherche d'information utilise la notion de *thème (about)*. Avant de modéliser le thème d'un bloc d'information, nous devons nous interroger sur la signification de ce thème, plus précisément la conception au niveau du document de l'*aboutness*[3]. Dans (Fairthorne, 1969) l'auteur distingue l'*aboutness* intentionnel et l'*aboutness* extensionnel. Le premier évoque la vue de l'auteur et ses intentions sur ce que contient le document. La seconde évoque ce que reflètent les unités sémantiques du document. Dans (Maron et Kuhns, 1960) les auteurs reprennent ces idées et déterminent l'*about* objectif qui exprime le lien entre un document et son index, et l'*about* subjectif qui définit le lien entre un document et la perception par l'utilisateur de ce document. Un bon index consiste alors en un index qui rapproche l'*about* objectif et l'*about* subjectif.

Le thème obtenu sur un passage ne correspond pas à un sens mais à une synthèse de plusieurs phrases. Nous émettons ici l'hypothèse, réaliste en recherche d'information, que le thème s'extrait à partir des régularités de l'ensemble des *sens* d'un bloc d'information. La création de la représentation d'un document peut donc s'effectuer en deux étapes :

- La première correspond à l'extraction du *sens* des phrases, c'est-à-dire la création des graphes représentant les phrases ; cette étape est inspirée de la théorie Sens Texte, même si les graphes utilisés ne correspondent pas à des sens.
- La seconde consiste en une étape de synthèse des graphes de phrase permettant de faire émerger le thème du document ; cette étape peut varier selon le modèle utilisé. Nous verrons dans cette thèse que les deux modèles proposés utilisent deux méthodes différentes.

Au final, la correspondance entre un document et une requête se fait au niveau du document ou du passage, et non pas par une correspondance de phrases.

3.2.2 Caractéristiques des représentations expressives

L'obtention de représentations expressives passe par l'utilisation du traitement de la langue. L'approche du traitement de la langue en recherche d'information diffère de celle des linguistes. Les approches traditionnelles de traitement de la langue privilégient la précision et la finesse de l'analyse linguistique (e.g. l'attachement prépositionnel, la désambiguïsation de sens, etc.). Ces approches s'évaluent en termes de précision et de complétude. Nous pensons qu'une telle précision devient inutile dans un contexte de recherche d'information. En effet cette tâche nécessite d'apparier des documents de manière globale sur des similarités de distributions statistiques, et non pas[4] de calculer le sens exact d'une phrase, comme dans la théorie Sens Texte, et d'en déduire un lien sémantique avec la requête. Cette tâche consiste plutôt à approximer le thème d'un bloc d'information. Les régularités statistiques au sein du bloc d'information (e.g. répétition, utilisation de synonymes) mettent en avant les éléments représentatifs et suppriment une part de l'incertitude sur les éléments extraits.

Cela représente l'approche classique en recherche d'information. Par exemple, dans un système vectoriel, une lemmatisation correcte des termes n'a pas d'impact fort. La plupart du temps, une *troncature*[5] qui réduit *'portique'* et *'porter'* en *'port'* reste tout à fait satisfaisante. En fait, le contexte d'usage du terme avec d'autres termes suffit à lui donner un rôle efficace pour discriminer des

[3] En effet, deux visions épistémologiques de l'*aboutness* se distinguent d'une part au niveau document, c'est la vision indexation, d'autre part la vision requête, c'est-à-dire la vision qui étudie la requête utilisateur pour voir l'indexation comme la solution de la requête.

[4] du moins dans l'état actuel de la recherche en recherche d'information

[5] ou 'stemming' en anglais.

documents. De la même manière, désambiguïser les termes pour associer un unique concept ne s'avère probablement pas si important. En effet, les concepts supplémentaires et erronés interfèrent peu avec le processus de correspondance à cause du contexte des autres termes de la requête. Au final, la couverture totale du document a plus d'importance que la résolution des ambiguïtés car une mauvaise résolution entraîne des problèmes de rappel (Sanderson, 1994) alors que leur non désambiguïsation a un impact négatif négligeable.

L'état actuel du traitement de la langue ne permet pas d'obtenir une représentation sémantique exacte. Cependant des outils peuvent extraire une partie du contenu avec plus ou moins d'imprécision. Nous nous orientons vers la création de représentations dont la complétude n'est pas certifiée. La couverture du contenu restant importante, nous proposons d'utiliser des surreprésentations du contenu des documents, même si une partie de cette représentation s'avère inexacte. Nous intégrons cette inexactitude à l'aide d'un score qui représente la confiance de détection d'un élément par le processus d'extraction. Cette approche peut paraître surprenante car nous ne pouvons pas garantir que la représentation obtenue obtienne *un sens* explicite, alors que nous pourrions partir d'un formalisme existant avec une sémantique explicite. Cependant est-ce un 'réel' problème en recherche d'information ? Est-ce que tout index produit par une machine doit obligatoirement avoir une signification pour un humain ?

Actuellement, les indexations les plus courantes se basent sur des ensembles ou des vecteurs de mots tronqués ou lemmatisés complétés de mesures statistiques, et personne ne pose la question de la *signification*[6] d'une telle structure. Pourquoi une représentation conceptuelle abstraite devrait avoir une signification ? Sans apporter de conclusion définitive, nous pensons qu'un index produit automatiquement n'a pas forcément à respecter la contrainte d'une dénotation.

3.2.3 Score de confiance

Actuellement la majorité des traitements qui aboutissent à la création d'une représentation d'une phrase ou d'un document ne produisent pas des structures totalement justes. Certains choix lors de la création d'une structure, notamment lors du traitement des ambiguïtés, s'effectuent arbitrairement ou en utilisant des informations statistiques : par exemple le choix de la structure la plus fréquente. Cependant, au final, la majorité des analyseurs fournissent un résultat unique masquant l'ensemble des choix effectués. Nous pensons au contraire que ces choix sont très importants et qu'ils doivent s'intégrer au sein même du modèle, notamment dans l'optique d'améliorer la précision. Faire correspondre un élément de la requête avec un élément du document dont la détection est certaine est meilleur que de faire correspondre ce même élément avec un élément ambigu ou incertain. Conserver un maximum d'informations concernant les sélections relatives au traitement de la langue permet une approche plus complète que d'utiliser seulement l'analyse unique proposée en fin de traitement.

Nous proposons de prendre en compte cette imprécision de l'analyse à l'aide d'un score de confiance sur les éléments du graphe. Ce score reflète la qualité et les choix du processus qui amène la détection d'un élément pour la représentation finale.

4 Mise en œuvre de la thèse

Pour mettre en œuvre les différentes propositions, la suite de ce manuscrit évalue à travers l'état de l'art de la partie 2 l'intérêt de l'expressivité en recherche d'information textuelle. Cet état de l'art permet d'étudier le niveau d'expressivité à adopter lors de la création d'un modèle orienté précision.

[6] Au sens de *dénotation* c'est-à-dire la portion de réalité que l'expression désigne

À la suite de cet état de l'art, nous proposons une modélisation qui met en avant l'expressivité, cela correspond à l'étape **A** sur figure 11. Cette modélisation décrite dans la troisième partie porte en premier lieu sur la modélisation de l'expressivité à travers la proposition d'un *cadre de modélisation* (**a** de la figure 11) qui utilise des supports de vocabulaires. Cette partie porte en second lieu sur l'utilisation de ce cadre pour développer *deux modèles d'expressivité forte* (**b** de la figure 11) basés sur des graphes : le modèle local et le modèle global. Nous distinguons ces deux modèles indépendamment de l'expressivité, en adoptant deux méthodes distinctes de l'utilisation des supports de vocabulaires. Ces deux modèles permettent d'explorer les capacités de notre cadre de modélisation et sont adaptés pour la recherche d'information orientée précision.

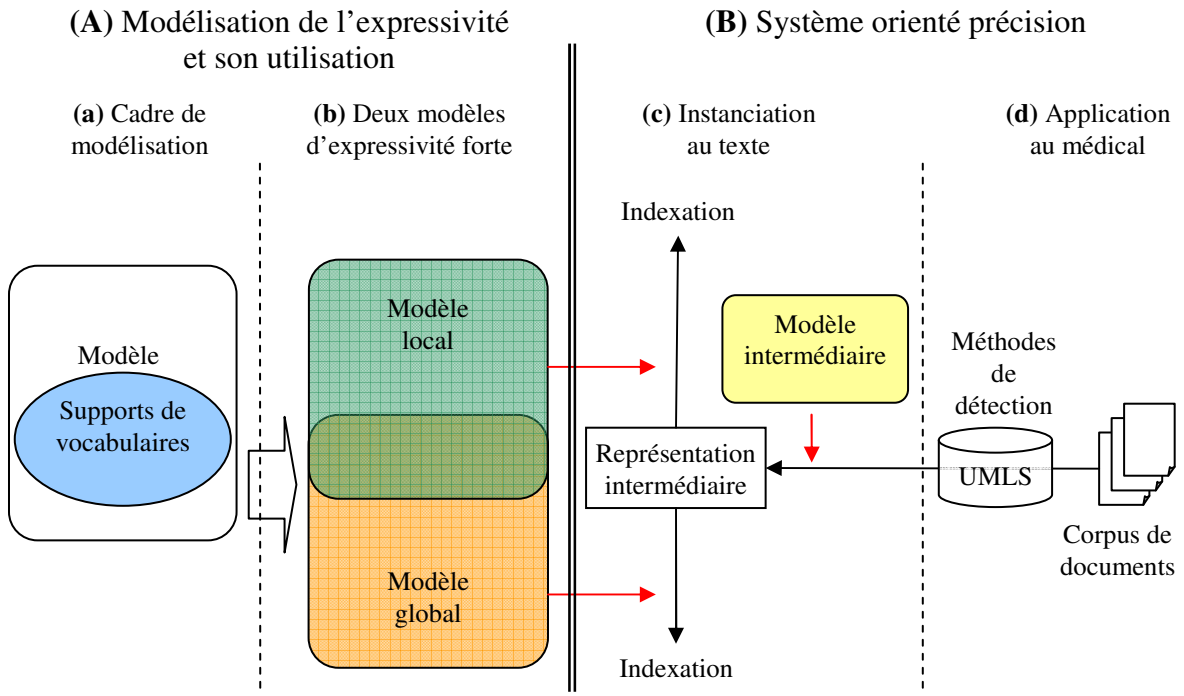


Figure 11 Schéma synoptique de la thèse

Une fois la modélisation présentée, nous décrivons dans la partie 4 les méthodes retenues pour développer sur le domaine médical les systèmes orientés précision qui correspondent aux deux modèles, cela correspond à l'étape **B** sur figure 11. Cette thèse propose d'utiliser un processus en deux étapes basé sur des traitements de la langue qui favorise la couverture des informations du document et qui utilise des scores de confiance pour rendre compte des imperfections d'un tel traitement. La première étape produit une représentation intermédiaire des documents dans laquelle chaque phrase est représentée par un graphe. Cette étape est dépendante du domaine d'application, ici le domaine médical (**d** de la figure 11). La deuxième étape crée les représentations finales des documents à partir de la représentation intermédiaire. Les méthodes proposées pour chaque modèle sont génériques au texte et peuvent se réutiliser sur différents domaines (**c** de la figure 11).

Nous évaluons dans la cinquième partie les différents processus proposés sur le domaine médical à l'aide d'une collection d'évaluation. L'évaluation est effectuée sur la représentation intermédiaire, puis sur les deux modèles : local et global. Ces expérimentations valident les méthodes proposées dans un cadre réel et inter-laboratoire où elles fournissent d'excellents résultats.

Enfin nous concluons et nous donnons les perspectives de ce travail dans la partie 6.

PARTIE 2 : ÉTAT DE L'ART

Introduction

Nous choisissons d'utiliser des systèmes orientés précision qui prennent en compte des descripteurs élaborés selon l'axe de l'expressivité des modèles de ces systèmes. Cet axe correspond à celui mis en avant par les supports de vocabulaires que nous proposons et se rapproche de l'axe utilisé dans la théorie Sens Texte. La figure 12 symbolise l'expressivité des modèles selon cet axe qui s'étend des langages à expressivité faible (ou langages simples) vers ceux à expressivité forte (langages complexes).

Nous présentons sur cet axe les familles de langages d'indexation utilisées en recherche d'information :

- Langages fondés sur des informations morphologiques : essentiellement les langages à base de mots-clefs.
- Langages fondés sur des informations syntaxiques : langages à base de syntagmes ou de structures syntaxiques.
- Langages fondés sur des informations sémantiques : langages à base de concepts, de structures sémantiques.
- Langages fondés sur des informations structurelles syntaxiques ou sémantiques : langages intégrant la dépendance.

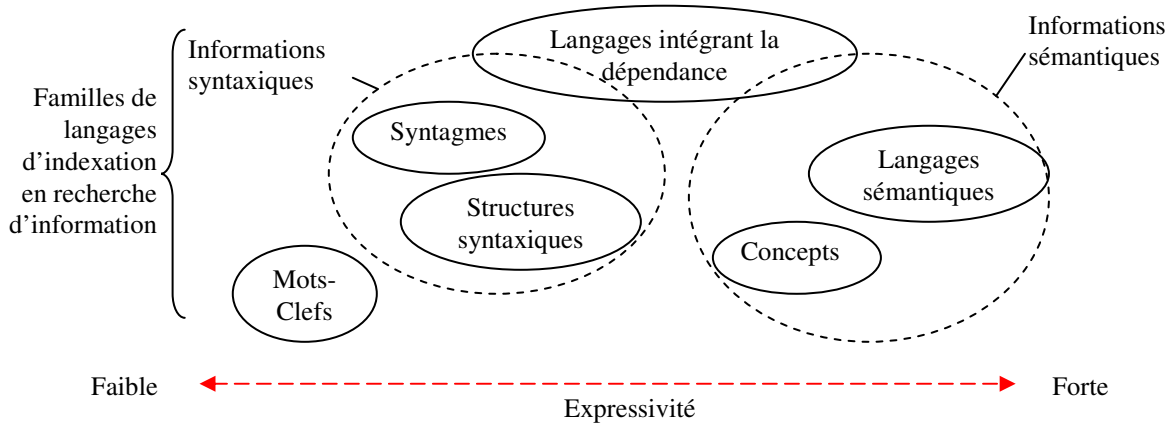


Figure 12 Axe représentant l'expressivité de la représentation de documents

Nous suivons cet axe pour présenter notre état de l'art.

Nous détaillons dans une première partie les systèmes qui intègrent des *informations syntaxiques* en recherche d'information : d'une part les moins expressifs qui se contentent d'ajouter des descripteurs à des représentations à base de mots-clefs et d'autre part les plus expressifs qui utilisent les structures syntaxiques pour comparer les documents et les requêtes.

Nous présentons ensuite des extensions des modèles de recherche d'information qui intègrent des *informations structurelles* syntaxiques ou sémantiques. Ces modèles de recherche d'information proposent des méthodes pour intégrer des structures. Ils s'appliquent à différents niveaux d'expressivité en fonction du niveau des structures sur lesquelles ils se basent.

Nous finissons en présentant un certain nombre d'approches qui intègrent des *informations sémantiques*. Parmi ces approches, nous détaillons plus précisément le modèle des graphes conceptuels qui permet de représenter les contenus sous forme de structures contenant des concepts et des relations sémantiques.

Chapitre III L'Information Syntaxique

« Le langage et l'outil sont l'expression de la même propriété de l'homme. » André Leroi-Gourhan (Le Geste et la parole)

Ce chapitre porte sur les systèmes qui intègrent des informations syntaxiques. De nombreux travaux s'intéressent à l'utilisation de telles informations du fait que les mots-clefs seuls ne capturent pas toutes les informations du texte. Les informations syntaxiques permettent la sélection dans la phrase de mots, de groupes de mots ou de structures plus complètes que les simples mots-clefs. De plus le traitement de la langue fournit un certain nombre d'approches et d'outils permettant d'extraire ces informations syntaxiques.

Après avoir présenté les deux structures syntaxiques les plus communes, nous détaillons leur utilisation en recherche d'information, puis leur application en question-réponse.

1 Les structures syntaxiques

Le niveau syntaxique, tel que défini en linguistique, correspond aux régularités de la langue spécifiques à la phrase. À ce niveau, les représentations modélisent les informations structurelles contenues dans les phrases telles que le rôle des mots et les relations syntaxiques qu'ils entretiennent. Les analyseurs qui produisent ces structures fournissent donc une structure par phrase. Deux types de structures syntaxiques couvrent la majorité des analyseurs utilisés. Elles correspondent à deux perspectives différentes de la structure de la phrase.

L'une exprime le plan catégoriel de la phrase : c'est l'analyse en constituants. Elle produit un arbre syntagmatique représentant la phrase. L'autre exprime le plan fonctionnel de la phrase, c'est l'analyse dépendancielle. Elle produit un arbre de dépendance pour chaque phrase. La théorie Sens Texte préconise cette dernière perspective. Les analyseurs correspondant à la première structure effectuent le marquage de *'groupes'*, ceux produisant la seconde s'intéressent plutôt aux *'relations'* entre les mots.

1.1 La structure syntagmatique

L'extraction de la structure syntagmatique de la phrase est effectuée à l'aide de grammaires de constituants. L'idée fondamentale de ces grammaires provient de la théorie des grammaires distributionnelles et repose sur le principe qu'une phrase est composée de groupes (les constituants), eux-mêmes structurés en sous-groupes, pour arriver à l'étape finale, les mots. L'analyse correspond donc à la détection de ces groupes par décomposition de chaque unité syntagmatique en unités de plus en plus petites, en partant de la phrase pour aboutir aux mots.

La structure syntagmatique se représente généralement par un arbre syntagmatique ou par un parenthésage étiqueté de la phrase. Dans la figure 13 nous présentons un exemple de représentation par l'arbre syntagmatique de la phrase *'Jupiter est la plus grande planète du système solaire'*.

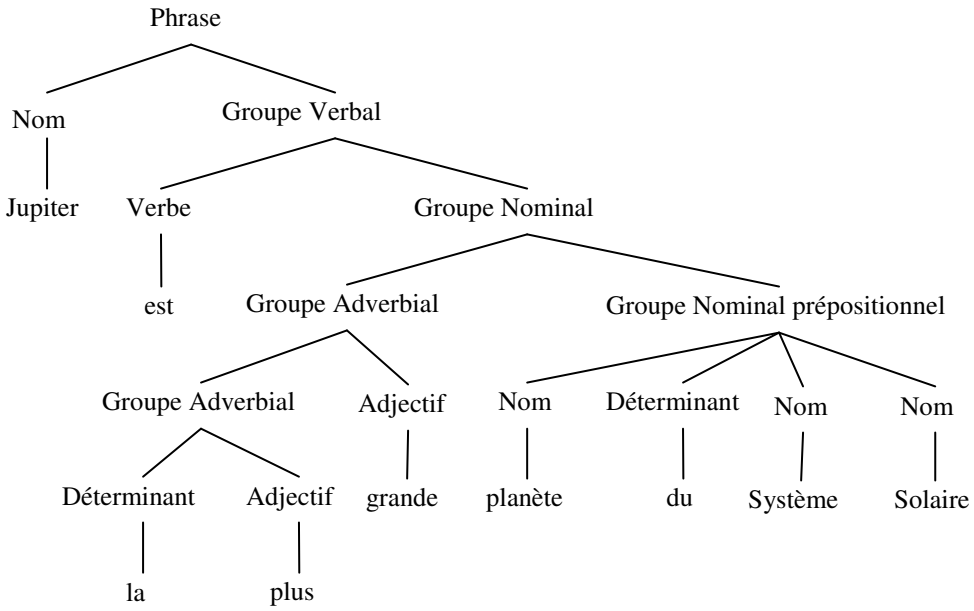


Figure 13 Arbre syntagmatique de la phrase 'Jupiter est la plus grande planète du Système Solaire'

Cette structure s'adapte facilement aux langues à ordre fixe comme l'anglais, mais ne peut pas s'appliquer à des langues à ordre variable telles que le finlandais ou le russe.

La prédominance des grammaires de constituants en linguistique a duré jusqu'aux années 70. Cette structure s'adapte parfaitement à la langue anglaise qui est restée longtemps prédominante en recherche linguistique.

1.2 La structure dépendancielle

La structure dépendancielle se base sur un modèle d'analyse syntaxique qui a pour but la description des connexions structurales entre les mots en tant qu'éléments constitutifs de la phrase. (Tesnière, 1959) introduit[7] cette structure en 1959 et Mel'čuk la reprend par la suite au sein de la théorie Sens Texte.

Dans une phrase, les mots ne font pas que se suivre, comme postulé par les grammaires de constituants, les mots entretiennent des relations. Les grammaires de dépendances mettent en évidence les relations établies entre les différents mots de la phrase (Sujet, COD, etc.). Celles-ci se définissent comme des dépendances, par exemple un sujet dépend d'un verbe. La suppression du verbe de la phrase supprime le sens du sujet qui n'a alors plus de raison d'être au sein de la phrase. Les dépendances font donc intervenir deux fonctions pour un mot vis-à-vis d'un autre mot : celle de **gouverneur** et celle de **dépendant**.

- Un mot **gouverne** un autre mot quand son apparition au sein de la phrase détermine celle de l'autre mot. Un mot peut gouverner plusieurs autres mots.
- Un mot **dépend** d'un autre quand l'apparition de ce mot dans la phrase dépend de ce second mot. Dans une phrase, chaque mot dépend d'un seul autre sauf la racine de la phrase.

[7] Ou plutôt les réintroduit, les dépendances étant depuis très longtemps utilisées pour décrire la grammaire.

La structure dépendancielle se représente à l'aide de liens entre les mots ; soit directement sur la phrase soit en utilisant une structure d'arbre, l'arbre de dépendance, comme présenté sur l'exemple de la figure 14.

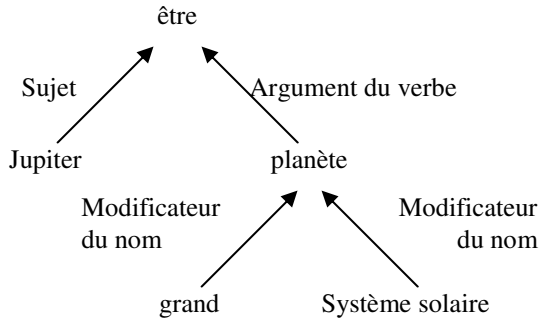


Figure 14 Arbre de dépendance entre mots pleins de la phrase 'Jupiter est la plus grande planète du Système Solaire'

Les grammaires de dépendance se rapprochent de l'analyse grammaticale traditionnelle ; le terme 'dépendance' se définit comme le strict synonyme de subordination. En cela, cette méthode se rapproche plus du mode humain de construction des phrases que la méthode syntagmatique.

1.3 Conclusion

Même si les deux méthodes d'analyse se différencient dans leur approche, des travaux plus récents mènent au rapprochement de ces structures, notamment par l'ajout de têtes aux syntagmes dans les grammaires de constituants, qui permet de retrouver les dépendances.

Les structures syntaxiques expriment le niveau d'analyse le plus bas d'une phrase qui fournit une structuration des informations de la phrase. Les analyses décrites ci-dessus correspondent à des analyses syntaxiques de surface car elles fournissent des représentations qui dépendent des structures de la langue analysée. Cependant ces analyses représentent la majorité des analyses fournies par le traitement de la langue.

2 Utilisation dans les systèmes de recherche d'information

En recherche d'information, différents travaux se fondent sur l'utilisation des informations syntaxiques. Ces études portent sur l'utilisation d'analyseurs syntaxiques ou morphosyntaxiques pour améliorer les performances des systèmes. La majorité de ces travaux vise à l'enrichissement de l'index à l'aide de descripteurs plus complexes que les simples mots-clefs. En effet, l'information syntaxique permet de détecter sur des textes des descripteurs tels que des mots composés, des couples de mots ou encore des syntagmes. D'autres travaux portent sur l'utilisation dans l'index de structures syntaxiques produites par les analyseurs, notamment la structure dépendancielle. Nous détaillons ces deux approches dans cette section.

2.1 Ajout de descripteurs

En recherche d'information, l'utilisation des mots composés se base sur le fait que ces éléments permettent de représenter le document plus précisément et plus complètement. En effet, les mots-clefs

ne représentent qu'une vue partielle d'un document. Des groupes de mots représentent mieux les entités sémantiques précises que de simples mots.

Par exemple, lors de l'utilisation de mots composés exocentriques tels que 'rouge gorge' les sens des unités lexicales ne permettent pas de reconstruire le sens du mot composé. En interrogeant un système basé uniquement sur les mots-clefs et pour une telle requête, les résultats obtenus peuvent porter aussi bien sur l'oiseau que sur les maux de gorge car ces deux types de documents contiennent les mots 'rouge' et 'gorge'. La prise en compte du syntagme comme terme d'indexation permet de différencier les documents où le terme 'rouge gorge' apparaît de ceux traitant de 'gorge rouge' et inversement. Utiliser les termes composés plutôt que les lexies qui les composent permet d'obtenir une meilleure représentation de l'information contenue dans un document.

Dans une phrase, un terme peut aussi se représenter par un groupe de mots. Utiliser une partie de la phrase, le plus souvent des groupes nominaux plutôt que des mots, améliore la recherche d'information car ils reflètent le thème de la phrase (Haddad et Chevallet, 2003). Certains systèmes complètent leurs index à l'aide de mots composés obtenus par des méthodes statistiques, telles que le comptage des mots adjacents dans la collection. D'autres travaux se basent sur l'intuition que les *syntagmes syntaxiques*, syntagmes dans lesquels les éléments entretiennent des relations syntaxiques, représentent le document de façon plus précise que les syntagmes obtenus statistiquement.

2.1.1 Extraction à l'aide d'étiquetage et de patron

Une première méthode pour extraire les mots composés, ou les syntagmes, consiste à utiliser un étiquetage morphosyntaxique puis à y appliquer des patrons syntaxiques. Mitra utilise notamment cette méthode (Mitra et al., 1997) et se sert d'un étiquetage des documents suivi d'une extraction des syntagmes nominaux pour reclasser les réponses d'un système à base de mots. Ses expérimentations ne montrent pas d'augmentations significatives par rapport à des résultats à base de mots. Suite à leurs observations, les auteurs concluent que les syntagmes n'améliorent pas significativement les premiers documents classés, mais qu'ils permettent de filtrer les documents peu pertinents. La méthode proposée se base sur un reclassement des 100 premiers documents retrouvés à l'aide des mots-clefs. Une partie des mauvais résultats imputés aux syntagmes provient de ce reclassement. Le système reclasse 100 documents, ceux où les mots-clefs apparaissent souvent mais pas obligatoirement ceux qui contiennent les syntagmes. Reclasser plus de documents permet de prendre en compte plus de syntagmes qui, en raison de leur structure, apparaissent plus rarement que les mots simples.

Dans (Haddad et Chevallet, 2003), les auteurs effectuent un étiquetage d'une collection en langue arabe. Un repérage de patrons extrait les syntagmes nominaux candidats pour l'indexation. Pour ne conserver que les syntagmes valides, les auteurs utilisent un filtrage. Ce filtrage favorise les syntagmes longs, ce qui permet de limiter leur nombre et permet ainsi d'obtenir des syntagmes complets. Cette utilisation des syntagmes, testée sur les collections de la campagne Amaryllis, fournit une amélioration entre 1.18 % et 6.05% sur la précision moyenne en fonction des collections.

2.1.2 Extraction à l'aide d'analyseur syntaxique

Les syntagmes peuvent aussi être extraits à l'aide du résultat d'un analyseur syntaxique en dépendance. Dans (Gaussier *et al.*, 2000), (Strzalkowski *et al.*, 1994), (Zhai *et al.*, 1997) les auteurs se basent sur l'hypothèse qu'un syntagme résulte de sa génération par une structure de dépendance. Une analyse des documents permet donc d'extraire les syntagmes. Zhai se sert d'un analyseur basé sur la maximisation de la vraisemblance. Strzalkowski extrait pour sa part une représentation proche d'un arbre de dépendance à l'aide d'un analyseur basé sur des grammaires. Enfin Gaussier effectue une analyse en dépendance sur des corpus en français.

Les auteurs sélectionnent ensuite un certain nombre de paires candidates à la formation de termes composés à l'aide de patrons sur la structure extraite ou en ne sélectionnant que les couples de mots pleins. Ils ajoutent ensuite ces descripteurs dans l'index des documents.

Strzalkowski propose un schéma de pondération sous forme de *tf.idf*, pondérant les termes par rapport à leur fréquence à l'intérieur du document (*tf*) et par rapport à leur fréquence documentaire inverse (*idf*)[8]. Le schéma permet de donner de l'importance à l'*idf* des termes composés. Les auteurs notent une augmentation de l'ordre de 20% de la précision moyenne avec l'utilisation des mots composés. Les expérimentations de Tong sur le système CLARIT montrent aussi une augmentation des performances de l'ordre de 20% sur TREC 3 et de 10 % sur TREC 5.

2.1.3 Conclusion

Si les approches qui ajoutent des descripteurs montrent des améliorations des résultats, celles-ci restent variables et dépendent fortement du contexte de la collection utilisée. Ces approches ne prennent que rarement en compte les caractéristiques propres aux descripteurs ajoutés. En effet, au niveau de la répartition des descripteurs, les syntagmes apparaissent moins que les mots. Les modèles proposés ne prennent pas toujours cela en compte. Ils ajoutent simplement les descripteurs à l'index en les considérant comme équivalents et même indépendants vis-à-vis des mots-clefs. Si certaines de ces approches conservent une part de la structure de la phrase, notamment celles qui extraient des descripteurs basés sur des dépendances, la majorité d'entre elles ne tiennent pas compte de la structure syntaxique de la phrase car elles décomposent cette structure.

2.2 Utilisation directe de la structure syntaxique

Partant de l'hypothèse que la conversion des structures de dépendance en syntagmes entraîne une perte d'informations, certains travaux portent sur l'utilisation complète de la structure de dépendance.

En effet, les liens entre les mots dans une phrase permettent de valider ou d'invalider des réponses à des requêtes. Dans la requête *'éducation par la recherche'* si le système ne prend pas en compte la structure de la phrase alors il apporte les réponses qui portent sur la recherche en éducation et non pas sur l'éducation *au moyen* de la recherche. Au contraire si le système type correctement le lien entre *'éducation'* et *'recherche'*, on élimine alors tous les documents ne contenant pas le bon lien. Pour une telle requête le système retrouve alors des documents plus précis.

Dans un système avec des mots-clefs et sans structure, les régularités statistiques valorisent le sens de cooccurrence le plus commun entre les éléments d'une requête. Dans le cas des mots *'manger'* et *'requin'*, c'est plus souvent le requin qui mange quelque chose qu'il n'est lui même mangé. La relation '*requin est sujet de manger'* est plus courante que les autres. La majorité des requêtes contenant *'requin'* et *'manger*' retrouvent donc ce schéma de cooccurrence entre ces deux mots. Si un utilisateur pose la requête *'manger du requin'* de nombreuses réponses ne vont pas être bonnes, car elles portent sur le schéma de cooccurrence le plus courant qui ne correspond pas à celui recherché. Utiliser la structure syntaxique pour interroger permet de sélectionner les bons schémas de cooccurrence.

De plus, la plupart des modèles intégrant des syntagmes utilisent le modèle vectoriel. Les syntagmes sont simplement attribués à de nouvelles dimensions du vecteur. Ces nouvelles dimensions dépendent des dimensions des mots qu'elles relient, une telle représentation ne capture pas correctement les liens entre les termes. Des approches proposent donc de prendre en compte la structure syntaxique, certaines utilisent partiellement la structure de la phrase, d'autres l'utilisent dans sa totalité.

[8].idf = log(N/n) où N est le nombre de documents dans le corpus, et n ceux que le terme indexe

2.2.1 Sur les Syntagmes

Certains travaux, plutôt que de traiter la structure de la phrase entière, se limitent à la structure des syntagmes (Ho, 2004). Le système utilise l'analyse syntaxique pour calculer la correspondance entre syntagmes structurés. Le but consiste à calculer la distance entre les syntagmes de la requête et ceux des documents. L'auteur considère le terme complexe comme l'élément central de l'indexation précise. Un terme se définit comme un élément d'une terminologie, c'est-à-dire un syntagme nominal qui désigne un concept unique dans un domaine donné (e.g. *'malformations vasculaires du système nerveux central'*).

La fonction de correspondance se complexifie car il ne s'agit plus seulement de comparer l'intersection pondérée de deux ensembles de termes (requêtes et documents), mais de comparer un ensemble de termes structurés.

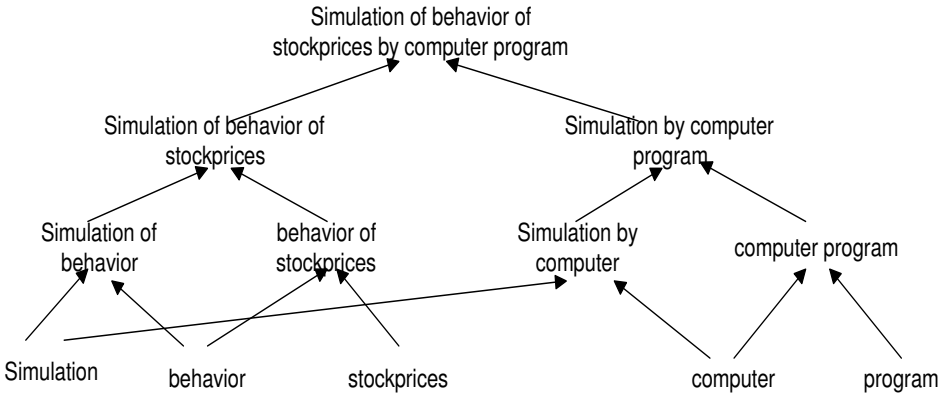


*Figure 15 Exemple de treillis de syntagmes (*Ho, 2004*)*

L'auteur propose une *correspondance par dérivation de termes* évaluée par un réseau Bayésien, extrait de l'arbre de dépendance, des syntagmes mis en relation (figure 15). Cette structure exprime les dérivations possibles d'un syntagme en un autre syntagme. Les résultats montrent que l'utilisation seule de ces termes d'indexation n'améliore pas les résultats de recherche d'information. L'auteur souligne que la qualité des syntagmes et la qualité de leur structuration dépendent de la qualité de l'analyse linguistique de surface. Cependant, l'auteur ne donne pas clairement la justification de l'interprétation probabiliste de son modèle qui se base sur une pondération en *tf.idf* pour les termes simples.

2.2.2 Sur les phrases

D'autres chercheurs utilisent la structure complète de la phrase. Dans (Matsumura *et al.*, 2000) l'auteur génère une structure proche des arbres de dépendances à partir de phrases en japonais. Dans sa structure, les nœuds terminaux représentent des 'mots concepts' comme les noms, les adjectifs ou les adverbes, et les autres nœuds représentent des 'mots relations' comme les verbes ou les particules post-positionnelles. Une analyse syntaxique utilisant les dépendances permet de générer des arbres sur les phrases correspondant aux titres et aux résumés des documents d'un corpus. Un ensemble d'arbres binaires constitue alors l'index des documents, les arbres correspondant aux structures syntaxiques produites sur les documents. De son côté, la requête se représente sous la forme d'un arbre binaire obtenu par le même processus.

La fonction de correspondance entre la requête et les documents se base sur une combinaison linéaire entre le score de *tf.idf* obtenu sur les documents et un score de dépendance. Ce score de

dépendance se calcule en deux étapes. La première effectue des découpages de l'arbre pour détecter les correspondances entre relations (cf. figure 16).

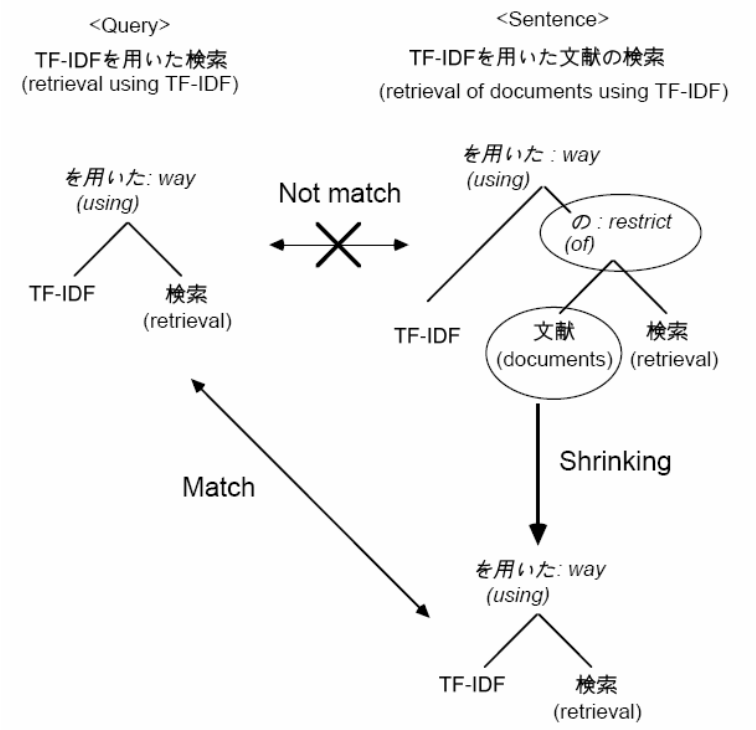


*Figure 16 Exemple de découpage de l'arbre (*Matsumura *et al.*, 2000*)*

La seconde applique une fonction de pondération basée sur un niveau de correspondance des dépendances et sur le poids des nœuds reliés. L'auteur distingue trois niveaux de correspondance :

- Exact : les deux mots reliés sont les mêmes.
- Catégorie : les mots sont différents mais les catégories sont les mêmes
- Externe : les mots et les relations sont différents

Le système attribue un score à chacun de ces niveaux et multiplie ce score avec la fréquence documentaire inverse des nœuds reliés par la dépendance.

L'auteur évalue cette méthode sur la collection NTCIR–1, en utilisant des résumés d'articles de conférences comme documents. Dans cette expérimentation, les auteurs déterminent que le score de correspondance exacte est égal au score de correspondance par catégories. Seul le dernier score (Externe) sert à caractériser les différents systèmes. Les résultats montrent que ce système est supérieur à une pondération *tf.idf* de 17% en précision moyenne, cependant une expérimentation complémentaire prenant en compte toutes les cooccurrences entre les mots donne des résultats similaires à 1% près, cela diminuant l'intérêt de la structure syntaxique.

Dans leurs travaux (Metzler et Haas, 1989) Metzler et Haas produisent des arbres de dépendances binaires sur des phrases en anglais. Les auteurs utilisent l'analyseur COP (Constituent Object Parser) pour produire ces arbres à partir des documents. Dans ce système, l'utilisateur doit déterminer les termes pertinents pour sa requête et indiquer les dépendances entre ces termes. Le système évalue alors les documents pertinents pour la requête en effectuant plusieurs types de correspondance entre les dépendances de la requête et celles contenues dans les arbres des documents.

Les deux analyses précédentes proposent une unique structure de la phrase qui ne prend pas en compte les ambiguïtés syntaxiques. Smeaton (Smeaton, 1999) propose un modèle appliqué sur les clauses, où chaque clause est représentée par l'arbre des ambiguïtés les plus courantes. La fonction de similarité s'effectue à l'aide d'une correspondance sur les arbres et par un regroupement des probabilités de chaque clause du document par rapport à la requête. Sur TREC-3, cette correspondance obtient cependant des résultats inférieurs à ceux obtenus en appliquant une pondération *tf.idf* sur les syntagmes représentés dans les arbres.

2.2.3 Conclusion

Ces approches prennent en compte la structure syntaxique pour améliorer les résultats et montrent, dans la majorité des cas, des résultats inférieurs à ceux basés sur les mots ou les termes. Même si ces études prennent en compte l'information de structure, peu d'entre elles s'appuient sur l'information syntaxique portée par les noms des relations syntaxiques. L'utilisation de la structure syntaxique complète entraîne aussi une augmentation de la complexité des traitements à effectuer lors de la correspondance entre la requête et les documents. Enfin dans ces approches, la correspondance entre le document et la requête s'effectue au niveau des phrases, or en recherche d'information, la recherche porte sur des documents entiers.

Dans nos travaux (Maisonnasse, 2005) nous proposons une approche intermédiaire où nous décomposons l'arbre de dépendance en un ensemble de triplets de dépendances prenant en compte la relation syntaxique. La combinaison de ces relations avec un index basé sur les lemmes montre une amélioration des résultats. Cependant ces résultats montrent également que ces augmentations sont trop dépendantes des langues et des types de requêtes.

2.3 Bilan de la syntaxe en recherche d'information

Si l'indexation à base de mots-clefs atteint des limites en recherche d'information, l'ajout de descripteurs basés sur la syntaxe n'apporte pas obligatoirement une amélioration des résultats. En effet, de tels descripteurs s'avèrent souvent trop précis et ces approches ne permettent pas de prendre en compte les variations terminologiques qui deviennent nombreuses quand la longueur des termes augmente, ce qui entraîne une diminution du rappel.

Même si certaines méthodes basées sur l'analyse syntaxique essayent de résoudre la variation de la structure, dans le cas général, elles restent trop proches de la forme de la phrase et ne permettent pas de prendre en compte toutes les variations d'une même expression.

Sur ces méthodes, de nombreux paramètres rendent difficile l'évaluation précise des résultats. Un facteur tel que la qualité de l'analyse a un impact sur les résultats finaux de recherche d'information. Or peu de travaux prennent en compte ou étudient la qualité de l'analyse utilisée.

Ces méthodes fournissent des résultats peu concluants au niveau global. Elles montrent cependant que l'information syntaxique améliore la précision des premières réponses.

3 Utilisation dans les systèmes de question-réponse

Contrairement aux systèmes de recherche d'information qui recherchent des documents portant sur un thème, les systèmes de question-réponse recherchent dans des documents une information précise permettant de répondre à une question. La majorité des systèmes de question-réponse actuels utilisent la même architecture. Une première étape consiste à déterminer le type de question et les éléments importants dans la question. Ensuite un filtrage basé sur un modèle de recherche d'information permet de déterminer les documents ou les passages pouvant contenir la réponse. Enfin une dernière étape

consiste à sélectionner la réponse dans ces passages. L'analyse syntaxique peut servir dans ces différentes étapes ; elle peut aussi servir à l'extraction dans la question des éléments pertinents, à l'extraction de la réponse ou encore elle peut être utilisée pour la recherche de passages. Nous nous intéressons essentiellement à ce dernier cas. En effet, en question-réponse l'étape de recherche d'information consiste à rechercher des unités (passages ou phrases) qui contiennent les éléments de la réponse, ce qui se rapproche de la tâche de recherche d'information. Lors de cette étape, sélectionner efficacement et précisément les bons passages permet de faciliter la détection de la réponse à la question. Cette tâche peut être effectuée en utilisant des informations syntaxiques. Par exemple, dans (Ferret *et al.*, 2001) le système de question-réponse utilise un filtrage des documents basé sur les variations terminologiques de termes. Certains travaux en question-réponse utilisent aussi la structure et les relations syntaxiques pour améliorer cette sélection. Nous présentons ces travaux dans la suite de cette section.

3.1 Sélectionner des relations

3.1.1 Sélection stricte

Certains travaux portent sur l'utilisation de correspondances entre les dépendances de la question et les dépendances des passages susceptibles de contenir la réponse. Dans le travail (Katz et Lin, 2003), les auteurs utilisent l'analyseur MiniPar pour analyser les documents de la collection. De l'analyse résultante, les auteurs effectuent une sélection des relations les plus appropriées, celles correctement extraites par l'analyseur, et ils génèrent à partir de cette analyse des relations plus sémantiques (cf. tableau 1).

Phrase	Relations extraites
L'oiseau mange le serpent	[oiseau mange serpent]
Le serpent mange l'oiseau	[serpent mange oiseau]
Le plus grand volcan de la planète	[volcan adjmod grand] [planète poss volcan]
Le volcan de la plus grande planète	[planète adjmod grand] [planète poss volcan]

Tableau 1 Exemple de relations sélectionnées (Katz et Lin, 2003)

Les auteurs indexent ensuite la collection à l'aide de l'ensemble des triplets de relations sélectionnés. Ils utilisent ces index relationnels pour rechercher les phrases les plus à même de contenir les éléments de réponse à une question. Les auteurs comparent enfin les résultats de cette méthode avec une sélection des phrases basée sur une recherche booléenne accompagnée d'un classement de ces phrases effectué sur la densité des mots significatifs de la question. Les résultats montrent une forte augmentation de la précision accompagnée d'une chute du rappel.

Ces résultats positifs obtenus sur la précision s'expliquent en partie par le corpus utilisé. Les auteurs construisent ce corpus spécialement pour démontrer l'intérêt des relations pour la résolution de certains problèmes linguistiques non résolus par des traitements habituels de recherche de passages. Un cas d'utilisation normale devrait donner des performances plus modérées.

3.1.2 Correspondance incertaine

Les systèmes qui effectuent des correspondances strictes entre les relations de la requête et celles du document, même s'ils permettent d'obtenir une forte précision, ne fournissent pas une assez bonne couverture des réponses. En effet, ces systèmes ne fonctionnent pas quand les relations exprimées dans la réponse sont sémantiquement équivalentes à celles de la question, mais que leur expression

syntaxique diffère. Pour résoudre ce problème, les travaux (Cui *et al.*, 2005) et (Lin et Pantel, 2001) proposent des méthodes de correspondance incertaine entre les dépendances de la requête et celles du passage réponse. Pour cela, toujours à l'aide de l'analyseur syntaxique en dépendance MiniPar, les auteurs détectent des chemins syntaxiques entre les mots. Un chemin syntaxique se définit comme l'ensemble des dépendances et des mots qui relient deux nœuds de la phrase dans l'arbre syntaxique (cf. tableau 2).

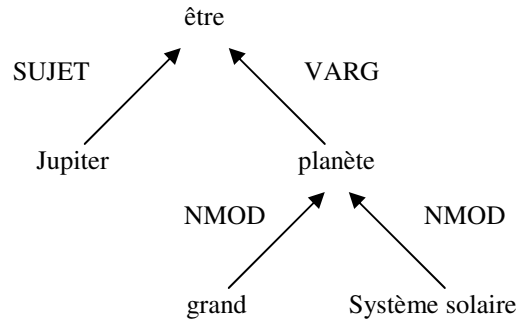
Arbre de dépendance	Chemin de dépendance extrait
	Jupiter SUJET être Jupiter SUJET VARG planète Jupiter SUJET VARG NMOD grand Jupiter SUJET VARG NMOD Système solaire être VARG planète être VARG NMOD grand être VARG NMOD Système solaire planète NMOD grand planète NMOD Système solaire grand NMOD NMOD Système solaire

Tableau 2 Ensemble des chemins extraits de 'Jupiter est la plus grande planète du Système Solaire'

Lin et Pantel (Lin et Pantel, 2001) proposent d'apprendre des relations entre termes, par exemple *'est auteur de'* à partir de ces chemins. Pour leur part, dans (Cui *et al.*, 2005), les auteurs proposent une correspondance entre les chemins de la requête et les chemins de chaque phrase d'un ensemble de documents préalablement sélectionné par un filtrage à base de mots-clefs. Ils définissent cette correspondance entre chemins comme non stricte : seules les extrémités du chemin, c'est-à-dire les nœuds, doivent correspondre. Cette correspondance se base sur un apprentissage sur un corpus de questions associées à des passages contenant les réponses. La similarité de deux chemins est calculée à l'aide de deux mesures statistiques et le corpus de TREC-12 permet de tester le système. Un ensemble de questions et de réponses extrait des précédentes campagnes TREC sert à apprendre les mesures statistiques. Ce système obtient une forte amélioration des résultats de la recherche de passages, de l'ordre de 80% par rapport à d'autres systèmes basés sur des densités de termes ou sur de la correspondance stricte de chemins de dépendance.

3.2 Distance entre arbres

D'autres travaux menés par Punyakanok (Punyakanok *et al.*, 2004) évaluent la distance entre la question et une phrase pouvant contenir la réponse par une correspondance floue entre leurs arbres de dépendance. Cette correspondance fournit une distance pour chaque couple d'arbres de dépendance. Cette méthode considère la phrase qui minimise la distance entre son arbre de dépendance et celui de la question comme celle qui répond le mieux aux attentes d'une question.

Le système évalue la distance par rapport au coût de la séquence de transformation minimale permettant de convertir un arbre ordonné et étiqueté en un autre arbre. Une séquence d'opérations peut contenir trois opérations : l'insertion ou la suppression d'un nœud et la substitution d'un nœud par un autre. A chacune de ces opérations correspond un coût. Le coût de la séquence minimale se calcule par la somme des coûts de chaque opération constituant la séquence.

Le tableau 3 présente la transformation de la phrase *'Saturne est la deuxième plus grosse planète du Système Solaire.'* en *'Jupiter est la plus grande planète du Système Solaire.'*. Cette transformation s'effectue en deux étapes : la suppression du nœud '*deuxième*' puis la substitution du nœud *'gros'* par

le nœud *'grand'*. Le coût de la transformation s'obtient en additionnant le coût de la suppression du nœud '*deuxième*' et le coût de la substitution de du nœud '*gros*' par '*grand*'.

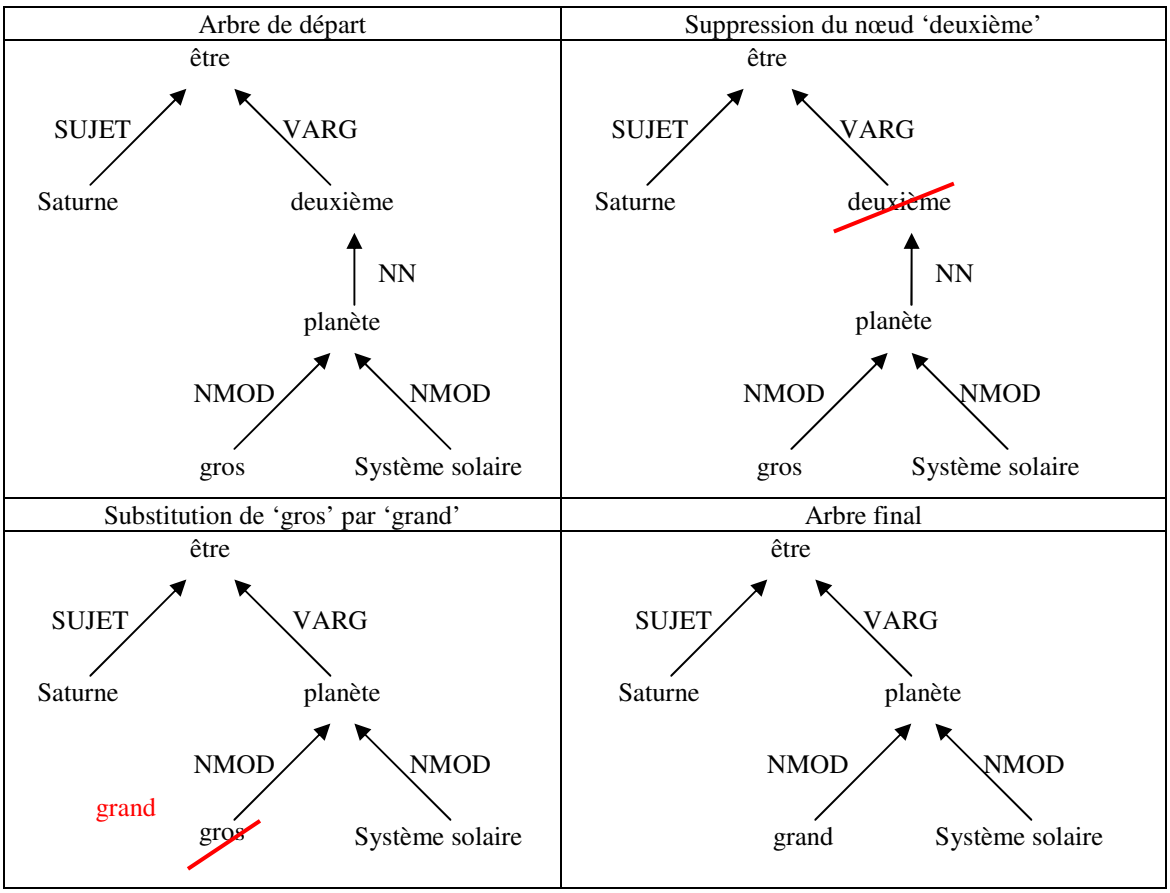


*Tableau 3 Transformation de la phrase 'Saturne est la deuxième plus grosse planète du Système Solaire.' en 'Jupiter est la plus grande planète du Système Solaire.' (*Punyakanok *et al.*, 2004*)*

Le calcul de la correspondance produit un ensemble de paires de nœuds permettant de faire le lien entre les deux arbres, cela en respectant un certain nombre de règles, notamment la conservation des parents et l'ordre des nœuds. Le coût de la correspondance s'évalue ensuite en calculant la somme des coûts de la transformation des nœuds de la correspondance et des coûts de suppression des nœuds de chaque partie de l'arbre qui n'appartiennent pas à cette correspondance.

Les auteurs évaluent cette méthode sur les 500 questions de la compétition TREC 2002. Pour chaque question, ils créent un ensemble de documents candidats élaboré à l'aide de l'ensemble des réponses (justes ou fausses) fournies par les participants de TREC 2002. Les auteurs sélectionnent, à l'aide de leur méthode et à l'aide d'une méthode basée sur les densités de termes, les documents censés contenir la réponse. La comparaison des résultats des deux méthodes montre que l'utilisation de la correspondance d'arbre améliore le nombre de documents corrects sélectionnés d'environ 40 %. Cependant cette méthode nécessite d'attribuer correctement un coût aux différentes opérations, ce qui la rend délicate à employer.

4 Conclusion

Différents travaux utilisent les informations syntaxiques. Le domaine du traitement de la langue fournit les analyseurs syntaxiques et ces analyseurs permettent ensuite de produire des informations syntaxiques à partir des documents. Ces travaux portent sur différentes utilisations des informations contenues dans la structure syntaxique, celle-ci allant du mot à la structure complète de la phrase.

En recherche d'information, ces travaux ont montré une amélioration des résultats, cependant cette amélioration dépend du domaine et du type de tâche réalisée. De plus les méthodes d'extraction basées sur des informations syntaxiques, telles que celles utilisées pour extraire les syntagmes, donnent dans la plupart des cas des résultats proches de ceux obtenus par des méthodes statistiques (Fagan, 1987)(Mitra *et al.*, 1997). Mitra soulève le fait que les syntagmes syntaxiques donnent de meilleurs résultats que les syntagmes statistiques lorsqu'un système utilise seulement les syntagmes. Cependant cet avantage disparaît si le système les regroupe avec les termes simples.

Même si l'utilisation de la syntaxe produit globalement de meilleurs résultats que les mots-clefs, notamment en ce qui concerne la précision, l'utilisation de la syntaxe reste insuffisante ou incomplète. D'une part, les analyseurs ne fournissent pas obligatoirement des informations exactes ; d'autre part, ces informations se rapprochent trop de la structure de la phrase, or plusieurs phrases peuvent avoir des structures différentes mais refléter les mêmes informations. Améliorer les résultats nécessite d'intégrer les variations de la structure, cela donne des résultats intéressants, notamment en question-réponse, mais reste à tester en recherche d'information.

De nombreux auteurs imputent une partie des faibles résultats dans les approches utilisant les structures syntaxiques à la mauvaise intégration de ces informations dans les systèmes de recherche d'information. En effet le *tf.idf* n'est pas totalement adapté à la pondération des syntagmes. De même ces approches ne prennent pas en compte le caractère relationnel des arbres.

Ainsi d'autres travaux s'intéressent à l'intégration d'informations structurelles au sein des modèles de recherche d'informations. Nous présentons ces approches dans le chapitre suivant.

Chapitre IV L'Information Structurelle

« Le plus grand danger des autoroutes de l'information, une fois de plus, ça sera pour les hérissons de l'information... » Jean-Marie Gourio (Brèves de comptoir)

La plupart des modèles de recherche d'information se basent sur l'hypothèse d'indépendance des termes. Cette hypothèse simplificatrice est clairement fausse du fait de la cooccurrence des termes et des liens syntaxiques qu'ils entretiennent dans une même phrase. Des recherches ont porté sur la prise en compte de cette interdépendance entre termes au sein des modèles, notamment pour améliorer leur précision. Une partie de ces modèles exploite la structure syntaxique des phrases pour prendre en compte les dépendances entre termes. Ces modèles, généralement génériques, peuvent s'appliquer à différentes représentations arborescentes de la phrase. Le modèle probabiliste et le modèle de langue sont deux modèles de recherche d'information dans lesquels la dépendance a été introduite formellement. Ce chapitre détaille successivement les extensions proposées pour ces deux modèles.

1 Extension du modèle probabiliste

1.1 Le modèle de base

Les modèles de recherche d'information modélisent la relation de pertinence entre un document et une requête. Dans le modèle probabiliste, l'hypothèse de base consiste à déterminer les probabilités $P(R|D)$ et $P(NR|D)$ pour une requête Q donnée pour ainsi évaluer la pertinence d'un document *D* par rapport à la requête. Ces deux probabilités déterminent la probabilité d'obtenir de l'information pertinente sachant un document : $P(R|D)$ et réciproquement, la probabilité d'obtenir de l'information non-pertinente sachant un document : $P(NR|D)$. Les documents peuvent alors se classer selon leur *rsv* (Relevance Status Value[9]):

$$rsv(D,Q) = \frac{P(R|D)}{P(NR|D)}$$

Ces deux probabilités ne peuvent pas s'estimer directement. En appliquant le théorème de Bayes pour simplifier ces estimations, la fonction se réécrit :

$$rsv(D,Q) = \frac{P(D|R)\frac{P(R)}{P(D)}}{P(D|NR)\frac{P(NR)}{P(D)}}$$

[9] Valeur de l'état de pertinence.

Sachant que les probabilités $P(R)$, $P(NR)$ et $P(D)$ sont des constantes pour une requête Q donnée, leur suppression n'affecte pas l'ordre des documents fourni par $rsv(D,Q)$. En recherche d'information les systèmes utilisent donc couramment une valeur de pertinence calculée comme suit :

$$rsv(D,Q) = \frac{P(D|R)}{P(D|NR)}$$

Pour estimer ces deux probabilités, $P(D|R)$ et $P(D|NR)$ les modèles décomposent les documents en des ensembles de sous-événements binaires $D = t_1, t_2, t_3, ...t_n$, un événement représentant la présence ou l'absence d'un mot. En émettant l'hypothèse d'indépendance des termes, la fonction précédente se réécrit :

$$rsv(D,Q) \approx rsv'(D,Q) = \prod_{j=1}^{n} \frac{P(t_i|R)}{P(t_i|NR)},$$

où :

$P(t_i|R)$ est la probabilité que t_i apparaisse dans un document pertinent,

$P(t_i|NR)$ est la probabilité que t_i apparaisse dans un document non pertinent.

Si le modèle considère les événements comme binaires $x_i=\{0,1\}$ et que l'événement t_i a la probabilité p_i dans les documents pertinents et q_i dans les documents non pertinents alors les probabilités des termes s'écrivent :

$$P(t_j = x_j|R) = p_j^{x_j} \times (1-p_j)^{1-x_j} \text{ et } P(t_j = x_j|NR) = q_j^{x_j} \times (1-q_j)^{1-x_j}$$

Par conséquent :

$$rsv'(D,Q) = \prod_{j=1}^{n} \frac{\left(p_j{}^{x_j} \times (1-p_j)^{1-x_j}\right)}{\left(q_j{}^{x_j} \times (1-q_j)^{1-x_j}\right)}$$

Enfin, en prenant le logarithme de cette formule et en supprimant les constantes, nous obtenons la formule suivante :

$$log(rsv'(D,Q)) = \sum_{j=1}^{n} x_j \, log\left(\frac{p_j \times (1-p_j)}{q_j \times (1-q_j)}\right)$$

Le modèle doit ensuite déterminer la probabilité des termes, soit par apprentissage, soit à l'aide d'estimations. L'apprentissage établit une loi de distribution à partir d'un échantillon de documents pertinents et non pertinents (Sparck Jones *et al.*, 2000). Les estimations se calculent typiquement selon des méthodes paramétriques ; le modèle suppose que la distribution des mots suit une certaine norme parmi les documents pertinents (et non-pertinents) (Robertson et Walker, 1994)(Robertson *et al.*, 1995).

1.2 Extension du modèle

Les modèles probabilistes classiques ne prennent pas en compte les dépendances entre les mots. Or il existe des dépendances entre les termes des documents quand la présence ou l'absence d'un terme

influe sur la présence ou l'absence d'autres termes. Par exemple, le choix d'un verbe dans une phrase influe sur les sujet possibles : '*aboyer'* possède souvent '*chien*' comme sujet, rarement *'chat'*.

Des extensions au modèle probabiliste prennent en compte ces dépendances. Malheureusement, si le modèle doit tenir compte de toutes ces dépendances, les calculs des probabilités $P(D|R)$ et de $P(D|NR)$ deviennent trop complexes, car il faut tenir compte pour un terme de l'ensemble des termes qui le précèdent.

En effet, si l'on considère le vecteur $D = t_1, t_2, t_3, ...t_n$ d'événements binaires, le problème consiste alors en l'estimation de la probabilité $P(t_1, t_2, t_3, ...t_n|R)$. Cela représente 2^n combinaisons possibles et par conséquent nécessite 2^n estimations. En prenant en compte toutes les dépendances, le système doit donc calculer :

$$P(t_1, t_2, t_3, ...t_n|R) = P(t_1|R)P(t_2|t_1R)P(t_3|t_1t_2R)...P(t_n|t_1...t_{n-1}R)$$

Évidemment, avec l'hypothèse d'indépendance entre ces éléments, comme supposé dans le modèle de base, l'estimation de la probabilité ne nécessite plus que *n* estimations.

Pour cette raison, certains travaux proposent une solution intermédiaire entre l'indépendance des événements et l'estimation de 2^n probabilités, notamment en sélectionnant les dépendances les plus pertinentes et en approximant la prise en compte de toutes les dépendances par une prise en compte partielle des dépendances les plus pertinentes. Par exemple, dans ses travaux, Losee (Losee, 1994) utilise une troncature de l'expansion de Bahadur Lazarsfeld pour réduire le degré de dépendance entre les termes. L'expansion de Bahadur Lazarsfeld calcule dans un premier temps les probabilités des termes sous l'hypothèse d'indépendance. La prise en compte des dépendances dans les probabilités s'effectue par leur multiplication à des facteurs de correction près. Un facteur de correction consiste en une série de facteurs individuels liés à chaque degré de dépendance. Si le calcul porte sur l'expansion complète alors le calcul fournit la probabilité exacte ; effectuer la troncature de cette expansion permet de réduire l'exactitude de celle-ci et d'obtenir une estimation par rapport au degré de dépendance escompté.

Dans ce travail, pour déterminer les dépendances, les auteurs utilisent un arbre de couverture maximum. La construction de cet arbre se base sur la maximisation de l'information mutuelle entre un élément de l'arbre et un élément en dessous, cela en contraignant par une fenêtre textuelle la distance entre deux mots liés par une dépendance.

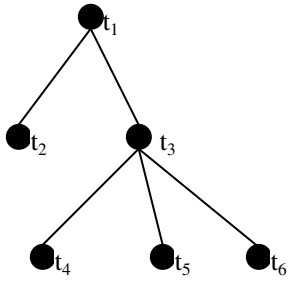


Figure 17 Maximum spanning tree ou arbre de dépendance

L'auteur évalue son approche à l'aide d'une collection constituée de 100 requêtes portant sur un ensemble de 1239 résumés. L'auteur évalue ses résultats sur cette collection à l'aide de la moyenne des longueurs de recherche. Les résultats montrent l'intérêt de la prise en compte des dépendances. De plus, par des variations sur la taille des fenêtres, l'auteur détermine que les dépendances les plus intéressantes relient des termes proches (entre 3 et 5 termes l'un de l'autre). Cette limitation provient

du fait que cette extraction s'effectue de manière statistique et sans apport d'informations. Même si cette méthode n'utilise pas d'informations syntaxiques, elle pourrait facilement se transposer pour utiliser de telles informations en remplaçant l'arbre de couverture maximum par un arbre de dépendance.

Lee et Lee (Lee et Lee, 2005) développent cette idée. Ils utilisent les dépendances syntaxiques extraites par leur propre analyseur sur du coréen. Ils incorporent ensuite ces dépendances dans un modèle probabiliste et adaptent ce modèle à l'aide de l'expansion de Chow. L'expansion de Chow (Chow et Liu, 1968) utilise initialement l''expected mutual information' pour créer un arbre de couverture maximum. Elle se base sur l'hypothèse suivante : le système peut sélectionner un certain nombre de variables de précédence $t_{j(i)}$ telles que $P(t_i | t_{i-1} \ldots t_j)$ dépend seulement de ces variables. Le calcul de la probabilité d'un élément peut alors se limiter à cet ensemble.

Par exemple : pour l'ensemble d'événements (t_1, t_2, t_3, t_4, t_5) et de précédences connues, tel que représenté par l'arbre de la figure 17, nous définissons la fonction *j(i)* qui associe à chaque terme t_i l'ensemble des éléments dont il dépend. Les probabilités des événements se calculent par :

$$P(t_2|t_1) = P(t_2|t_{j(2)}) = P(t_2|t_1)$$
$$P(t_3|t_1,t_2) = P(t_3|t_{j(3)}) = P(t_3|t_1)$$
$$P(t_4|t_1,t_2,t_3) = P(t_4|t_{j(4)}) = P(t_4|t_3)$$
$$P(t_5|t_1,t_2,t_3,t_4) = P(t_5|t_{j(5)}) = P(t_5|t_3)$$
$$P(t_6|t_1,t_2,t_3,t_4,t_5) = P(t_6|t_{j(6)}) = P(t_6|t_3)$$

La probabilité de l'ensemble d'événements de la figure 17 s'écrit alors :

$$P(D_i) = P(t_1)P(t_2|t_{j(2)})P(t_3|t_{j(3)})..P(t_n|t_{j(n)}).$$

Dans (Lee et Lee, 2005) les auteurs reprennent cette méthode mais remplacent l'arbre de couverture maximum par le résultat de l'analyseur en dépendance. Ils intègrent ensuite l'expansion de Chow au calcul de la probabilité qu'un document soit pertinent et à celle qu'un document ne soit pas pertinent. Ils adaptent ce calcul de probabilité pour qu'il puisse être utilisé sans information de pertinence.

Les auteurs évaluent ce modèle sur une collection qui se compose des 23 113 entrées d'une encyclopédie coréenne sur lesquelles 46 requêtes en langue naturelle sont évaluées. Les résultats comparant ce modèle par rapport à des modèles basiques montrent une amélioration d'environ 4% de la précision moyenne par l'utilisation des dépendances syntaxiques et de la troncature.

1.3 Conclusion

L'intérêt de ces modèles vient du fait qu'ils permettent de prendre en compte les relations entre termes. Cependant l'utilisation du modèle probabiliste limite ces modèles à la décomposition des probabilités de pertinence et de non pertinence. Le modèle probabiliste rend difficile l'intégration des relations en tant qu'événement du modèle et plus difficile encore l'utilisation de modèles où la structure ne contient pas de relations de dépendance.

2 Extension du modèle de langue

2.1 Le modèle de base

Ponte et Croft (Ponte et Croft, 1998) introduisent un modèle de langue appliqué à la recherche d'information en 1998. Les auteurs construisent un modèle où le score assigné à un document pour une requête se détermine par la probabilité qu'un modèle du document génère la requête. La différence majeure avec les modèles probabilistes vient du fait que le modèle ne pose aucune hypothèse de distribution et que modèle n'effectue pas de classification des documents en deux catégories : pertinents et non pertinents. Ce modèle remplace la probabilité de pertinence par la probabilité de produire la requête à partir du modèle de langue du document.

Ces modèles de langue s'inspirent des modèles utilisés dans le traitement de la langue, notamment en reconnaissance de la parole où un modèle de langue représente une distribution probabiliste qui capture les régularités statistiques d'une langue. Les systèmes utilisent ensuite cette distribution probabiliste pour la génération ou la reconnaissance de textes.

2.1.1 En traitement de la langue

Un modèle de langue *M* permet de calculer la probabilité $P(s|M)$ de toute séquence d'événements *s*. Cette fonction détermine la probabilité que le modèle de langue *M* génère la séquence. Si le modèle considère que chaque événement de la chaîne *s* dépend de l'ensemble des événements qui le précèdent alors $P(s|M)$ s'exprime selon :

$$P(s|M)=\prod_{i=1}^{n} P(e_i|e_1 \ldots e_{i-1}, M) \text{ avec } s=e_1 \ldots e_n$$

Mathématiquement, ce modèle se révèle complexe à mettre en place car il nécessite un grand nombre d'estimations. Pour réduire cette complexité, communément, les modèles posent l'hypothèse qu'un événement ne dépend que d'un certain nombre d'événements précédents :

$$P(e_i|e_1 \ldots e_{i-1}, M) = P(e_i|e_{i-1} \ldots e_{(i-p+1)}, M),$$

où *p* représente le nombre de prédécesseurs pris en compte.

En réduisant le nombre de prédécesseurs au minimum, modèle unigramme (*p=0*), on obtient l'hypothèse d'indépendance des termes habituellement utilisée en recherche d'information, un événement ne dépend alors d'aucun autre. En traitement de la langue, l'estimation de ces probabilités s'effectue à l'aide d'un corpus de texte assez grand, de manière à ce qu'il reflète la langue. L'estimation la plus courante utilise le maximum de vraisemblance, où la fréquence d'un N-gram *n* se calcule par :

$$P(n|M)=\frac{|n|}{|C|}$$

où $|C|$ représente le nombre d'éléments de même longueur que *n* dans la collection, et $|n|$ représente la fréquence de *n* au sein de la collection.

Cependant, le risque qu'un mot ou qu'une séquence, n'apparaisse pas dans ce corpus est important. Par l'utilisation d'estimations simples, l'absence d'un élément dans le corpus d'apprentissage se traduit par une probabilité nulle attribuée à cet élément. Si une séquence contient un mot qui ne se

trouve pas dans la collection, la probabilité de cette séquence devient alors nulle elle aussi. De manière à assouplir cette contrainte, les modèles assignent une probabilité non nulle aux éléments non rencontrés dans le corpus. Cette procédure se nomme le lissage. Il existe plusieurs méthodes de lissage (Boughanem *et al.*, 2004), nous présentons ici deux lissages souvent repris en recherche d'information et que nous utilisons dans la suite.

2.1.1.1 Lissage de Dirichlet

Ce modèle modifie directement le calcul de la probabilité d'un élément en modifiant sa fréquence à l'aide de sa probabilité dans un modèle d'ordre supérieur, pour cela la fréquence est modifiée à l'aide d'un paramètre μ :

$$P\left(e\middle|M_{final}\right)=\frac{tf\left(e,corpus\right)+\mu P\left(e\middle|M_{anglais}\right)}{\left|corpus\right|+\mu}$$

où $\left|corpus\right|$ représente la taille du corpus et $tf\left(e,corpus\right)$ représente la fréquence de *e* dans ce corpus. Ce modèle incrémente la fréquence de *e* de $\mu P\left(e\middle|M_{anglais}\right)$ qui s'appelle la pseudo fréquence, calculée ici pour sur corpus anglais.

2.1.1.2 Lissage de Jeliner-Mercer

Ce lissage effectue une interpolation entre un modèle de langue et un modèle d'ordre supérieur, ou plus large, de manière à éviter les probabilités nulles. La probabilité d'un élément se divise en deux contributions, le paramètre de lissage λ déterminant l'impact de ces contributions.

$$P\left(e\middle|M_{final}\right)=\lambda\times P\left(e\middle|M_{corpus}\right)+(1-\lambda)\times P\left(e\middle|M_{anglais}\right)$$

Dans cet exemple, le modèle final s'obtient en mixant le modèle de langue construit à partir d'un corpus en anglais avec un modèle de langue plus général de l'anglais. On obtient alors le modèle de langue final défini comme une variation du modèle de langue de l'anglais adapté au corpus. Nous utilisons ce lissage dans nos expérimentations basées sur les modèles de langue.

2.2 En recherche d'information

Un modèle de langue considère qu'un document *D* incarne une sous-langue, pour laquelle il construit un modèle de langue M_D. Ce modèle évalue ensuite la requête par rapport à cette sous-langue. L'idée à la base de cette approche est la suivante : un utilisateur qui rédige une requête le fait par rapport aux termes qui devraient apparaître dans les documents qui l'intéressent. Il sélectionne ensuite les termes capables de distinguer ces documents pertinents dans la collection (Ponte et Croft, 1998). L'utilisateur crée donc sa requête en fonction d'un modèle de document idéal. Le système doit alors comparer le modèle qui a abouti à la création de la requête à celui qui est à l'origine d'un document. Plus ces deux modèles se ressemblent, plus la pertinence du document pour l'utilisateur augmente (cf. figure 18).

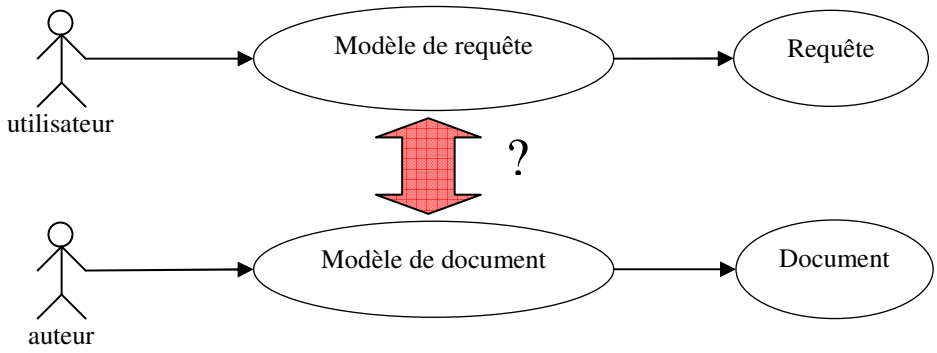
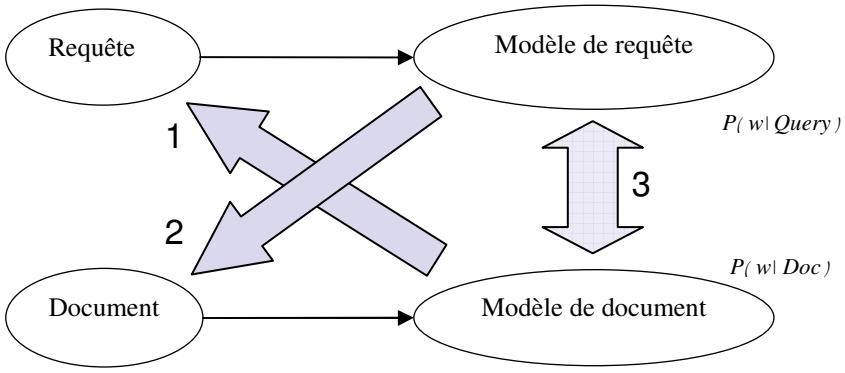


*Figure 18 Comparaison des modèles de (*Bruce Croft, 2002*)*

Les premiers modèles de langue s'appuient sur le fait suivant : plus le modèle d'un document a de probabilité de générer une requête, plus il se rapproche du modèle idéal de l'utilisateur, et par conséquent la pertinence du document pour la requête augmente. Le calcul de la *rsv* d'un document par rapport à une requête *Q* dépend donc de la probabilité que le modèle M_D génère cette requête.

$$rsv(Q,D) = P(Q \mid M_D)$$

Ce modèle calcule un score de vraisemblance de la requête. D'autres travaux explorent d'autres possibilités, en particulier la probabilité d'un document sachant le modèle de la requête (vraisemblance du document). D'autres encore comparent la modélisation de la requête et celle du document (Zhai et Lafferty, 2001). Ces possibilités sont résumées dans la figure 19.



Vraisemblance de la requête (1), Vraisemblance du document (2), Comparaison de Modèles (3)

*Figure 19 Méthodes de comparaison de la requête et du document dans les modèles de langue (*Bruce Croft, 2002*)*

Nous détaillons ici la vraisemblance de la requête, méthode la plus communément utilisée, et celle que nous utilisons dans nos expérimentations. Le système modélise la requête et le document comme des événements composés de plusieurs événements simples. Une requête de taille *n* se représente par une séquence de *n* variables $t_1,t_2,\ldots,t_n$. La probabilité de générer la requête se calcule alors par :

$$P(Q \mid M_D) = P(t_1, t_2, \ldots, t_n \mid M_D)$$

La séquence représentant le document constitue un échantillon d'un modèle de langue et sert de base pour la création du modèle. Du fait de la modélisation des documents, les fréquences telles que le *tf* ou *idf*, et la taille des documents font partie intégrante du modèle.

Lorsque l'on construit un modèle de langue d'un document, le principal inconvénient provient de la sous-représentation des données d'entraînement. Le document a une taille largement insuffisante pour estimer correctement les probabilités par des statistiques. De plus, le risque d'absence d'un terme dans le corpus d'apprentissage devient très fort et par conséquent le risque qu'une requête se voit attribuer un score nul pour un document augmente d'autant. Le lissage qui permet d'éviter ces problèmes, devient alors un élément très important.

Des travaux proposent plusieurs modèles de langue pour prendre en compte cette spécificité. (Ponte et Croft, 1998) utilise une estimation basée sur une fonction de risque. (Hiemstra, 1998) introduit un lissage entre le modèle local du document et un modèle global de la collection. (Berger et Lafferty, 1999) propose un modèle qui rajoute une étape de translation statistique.

2.3 Les extensions du modèle

Au contraire des autres modèles, les modèles de langue sont beaucoup plus propices à l'intégration de dépendances et d'informations syntaxiques. Les N-grams capturent une partie de ces dépendances : celles entre les termes d'une même fenêtre. Cependant, elles ne reflètent pas obligatoirement celles qui sont pertinentes pour l'utilisateur. Plusieurs travaux de recherche explorent la possibilité d'intégrer dans un modèle de langue des dépendances autres que la simple co-location.

2.3.1 Prise en compte des mots composés

Certains travaux portent sur l'incorporation de mots composés au sein des modèles de langue. Une méthode simple pour capturer en partie des groupes de mots ou des syntagmes consiste en l'utilisation de N-grams (Srikanth et Srihari, 2002). Ces N-grams ne capturent que l'ensemble des couples de mots consécutifs d'un document, il n'y a pas d'évaluation de la validité des groupes de mots sélectionnés. Il est clair qu'en prenant tous les N-grams possibles, de nombreux couples ne correspondent à aucun mot composé et n'ont donc aucun sens.

Par exemple dans le paragraphe ci-dessus on trouve le bigramme *'capturer en'* qui n'apporte aucune information sur le thème de ce paragraphe. Les modèles de langue N-grams font de plus l'hypothèse que l'ordre des mots est important. Cette hypothèse est correcte en général. En revanche, si les mots ne sont pas consécutifs, leur ordonnancement n'a pas la même importance. Par exemple, le couple *(recherche, information)* détecté dans la phrase *'la recherche du coupable s'appuie sur des informations produites par des témoins'* n'a strictement rien à voir avec le terme *'recherche d'information'*. Pour cette raison, des travaux portent sur l'étude de modèles de langue qui incorporent des paires de mots sur d'autres hypothèses que leur simple adjacence.

Dans (Alvarez *et al.*, 2004), les auteurs proposent un modèle de langue basé sur les affinités lexicales. Ces dernières sont définies comme des unités lexicales n'ayant pas de contrainte d'ordre, mais seulement une contrainte de distance. Le système sélectionne alors les affinités lexicales en fonction de leur *force* dans le document. Ils introduisent ensuite les paires sélectionnées dans un modèle unigramme en tant que nouveaux événements. Les résultats obtenus montrent de légères augmentations par rapport au modèle unigramme ou bigramme standard.

2.3.2 Extraire des concepts

Dans (Srikanth et Srihari, 2003) les auteurs considèrent une requête comme un ensemble de concepts. Ils supposent qu'identifier correctement les concepts qui représentent le besoin

d'information de l'utilisateur permet d'obtenir un modèle plus pertinent que des modèles manipulant de simples mots. Les auteurs supposent que les concepts apparaissent dans la requête au travers des syntagmes et qu'ils peuvent s'identifier par une analyse syntaxique. Ils considèrent finalement une requête comme une séquence de concepts, chaque concept se représentant lui-même par une séquence de termes. En appliquant l'hypothèse d'indépendance sur les concepts, les auteurs obtiennent alors la probabilité suivante :

$$P(Q|M_D) = \prod_i P(c_i|M_D) \text{ avec } c_i \text{ un concept}$$

Le modèle exprime la probabilité unigramme d'un concept par la probabilité jointe des termes de la requête exprimant le concept. La correspondance entre les termes et un concept s'effectue à l'aide de l'analyse syntaxique notamment en faisant correspondre un concept à des syntagmes. Cependant le système effectue une analyse sémantique seulement sur les requêtes. Par conséquent, le modèle évalue l'estimation des probabilités d'un concept à l'aide d'informations sur les cooccurrences des termes qui le composent :

$$P(c_i|M_D) \approx \prod_{l=1}^{n_i} P(t_{li}|t_{l-1\,i}, M_D) \text{ avec } t_{li} \text{ un terme formant le concept } c_i$$

La probabilité d'un terme sachant le modèle de langue M_D et son prédécesseur dans le concept se calcule par une interpolation entre la probabilité du bigramme et la probabilité du terme au sein du modèle de document

Les auteurs évaluent ensuite ce modèle sur différents corpus de TREC. La définition des concepts se limite aux syntagmes nominaux et verbaux. Les résultats montrent des augmentations entre 1 et 10 % sur la précision moyenne par rapport à des modèles unigrammes simples. Cependant, les résultats restent inférieurs à d'autres modèles de langue plus complexes.

Cette approche est intéressante puisqu'il s'agit d'une approche intermédiaire qui utilise les dépendances entre termes pour construire des concepts. Ces concepts représentent des événements spéciaux, constitués de termes.

2.3.3 Réduire le calcul des probabilités

Le calcul réel de la probabilité d'une phrase constitue un calcul complexe voire impossible car chaque mot dépend de tous les mots déjà apparus. Certains travaux limitent ainsi le nombre de dépendances en capturant les dépendances les plus fortes au niveau des phrases. La probabilité d'un terme ne se calcule alors plus en fonction de tous ses antécédents mais en fonction de son antécédent le plus probable, celui qui maximise la probabilité :

$$P(t_i|t_j, M) \text{ avec } t_i \text{ un terme et } t_j \text{ un de ces antécédents}$$

Dans (Lee *et al.*, 2006), les auteurs réutilisent l'expansion de Chow alliée à un arbre de dépendance afin de l'appliquer aux modèles de langue. Les modèles de langue cherchent à établir $P(q|M_d)$. La probabilité $P(q|M_d)$ s'écrit alors :

$$P(Q|M_D) = P(t_1|M_D)P(t_2|t_{j(2)}, M_D)P(t_3|t_{j(3)}, M_D)..P(t_n|t_{j(n)}, M_D)$$

Pour sélectionner les antécédents d'un terme, les auteurs utilisent les résultats d'une analyse par MiniPar. Pour un document ou une requête, l'arbre de dépendance permet d'établir l'ensemble des antécédents du terme $i : j(i)$.

Deux modèles de lissage sont ensuite testés : celui de Jeliner-Mercer et celui de Dirichlet. Les expérimentations sont conduites sur deux collections, TREC AP88 et WSJ90-92, avec les requêtes de TREC4. Un modèle bigramme avec les mêmes lissages permet de comparer les résultats. Globalement la méthode fournit des résultats supérieurs à ceux obtenus avec les modèles bigrammes.

Dans (Nallapati et Allan, 2002), la liste des antécédents de chaque mot est obtenue à l'aide d'un arbre de couverture maximum sur le degré de dépendance entre les termes de la requête. Le degré de dépendance entre deux termes au sein d'une phrase est estimé par le calcul de leur coefficient de Jaccard au niveau du document. La probabilité de produire une phrase sachant un modèle de document s'estime alors par le produit des probabilités de générer les termes sachant leurs antécédents dans l'arbre de couverture maximum.

Un lissage de la probabilité initiale avec un modèle de langue de l'anglais donne la probabilité finale. Le score d'un document pour une requête s'obtient par un rapport de vraisemblance : le rapport entre la probabilité que le modèle du document produise la requête sur la probabilité que le modèle de l'anglais la produise. Enfin, en parallèle, le modèle final s'obtient en mixant le modèle précédent avec un modèle unigramme. Les résultats montrent qu'un modèle unigramme donne de meilleurs résultats que le modèle basé uniquement sur les dépendances. Cependant les performances s'améliorent avec l'utilisation d'un modèle hybride qui combine les deux modèles.

Ces travaux indiquent que les dépendances syntaxiques peuvent s'intégrer en complément des unigrammes dans un modèle de langue. Cette intégration améliore les résultats, notamment ceux des bigrammes. Ces modèles utilisent cependant la dépendance uniquement pour réduire la probabilité réelle des mots et ils ne prennent pas en compte véritablement les dépendances comme une structure complète sur la phrase.

2.3.4 Génération de la requête basée sur une structure

Dans une autre approche (Gao *et al.*, 2004), les auteurs intègrent la structure de dépendance dans le modèle de langue. Le modèle de document créé n'intègre pas seulement les mots mais prend aussi en compte les dépendances entre ces mots. Par conséquent, le calcul des probabilités pour la requête s'en trouve modifié. Les auteurs considèrent les liens entre les termes comme une variable cachée qui permet d'exprimer les dépendances. L'approche fait l'hypothèse que la génération de la requête s'effectue par un processus en deux étapes :

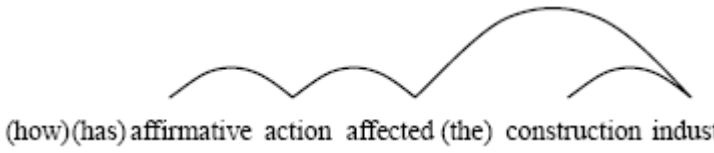


*Figure 20 Exemple de requête avec une structure L et les mots outils entre parenthèses (*Gao *et al.*, 2004*)*

- premièrement, le modèle produit la structure de dépendance *L* selon une probabilité $P(L|M_D)$ (cf. figure 20) ;
- ensuite le modèle génère la requête *Q* selon $P(Q|L, M_D)$. Cette étape choisit les termes de la requête en fonction des éléments liés dans *L.*

Dans ce modèle, la probabilité de produire la requête $P(Q| M_D)$ sachant toutes les structures de dépendance possibles *Ls* s'écrit :

$$P(Q|M_D) = \sum_{Ls} P(Q, L|M_D) = \sum_{Ls} P(L|M_D) P(Q|L, M_D)$$

Les auteurs supposent ensuite qu'une unique structure *L*, la structure la plus probable, domine l'ensemble des structures de dépendance possibles pour une phrase *Ls*. L'équation s'écrit alors :

$$P(Q|M_D) = P(L|M_D) P(Q|L, M_D)$$

Ils supposent que dans la structure *L* de la requête, chaque terme généré dépend uniquement d'un seul terme déjà produit. Ils supposent également que la structure *L* est acyclique et que chaque terme n'a qu'un seul gouverneur, à l'exception de la tête de la phrase. Les auteurs arrivent finalement à la décomposition suivante:

$$\log\left(P(Q|M_D)\right) = \log\left(P(L|M_D)\right) + \sum_{i=1..m} \log\left(P(q_i|M_D)\right) + \sum_{(i,j)\in L} MI\left(q_i, q_j|L, M_D\right)$$

où $MI(q_i, q_j|L, M_D) = \log \frac{P(q_i, q_j|L, M_D)}{P(q_i|D)P(q_j|M_D)}$

Le calcul de cette probabilité se base ensuite sur l'estimation de ses différents composants. L'estimation $P(q_i|M_D)$ se calcule à l'aide d'un lissage en deux étapes utilisant les lissages de Dirichlet et de Jelinek-Mercer auxquels les auteurs ajoutent un lissage pour le calcul des éléments de la collection.

L'estimation de $P(L|M_D)$ se base sur l'existence d'un corpus d'entraînement avec la structure définie pour chaque phrase ; ici il s'agit des documents de la collection. Les auteurs estiment alors la probabilité d'une structure *L* par la fréquence relative de deux éléments liés sachant qu'ils appartiennent à la même phrase.

$$P(L|Q) = \prod_{(i,j)} \frac{N(q_i, q_j, R)}{N(q_i, q_j)}$$

où

$N(q_i, q_j, R)$ représente le nombre d'apparitions de q_i et q_j reliés par une relation *R* de *L*.

$N(q_i, q_j)$ représente le nombre d'apparitions de q_i et q_j dans une même phrase.

Cette probabilité est tout d'abord lissée par un lissage de Jelinek-Mercer puis par un deuxième lissage qui prend en compte la probabilité de trouver des relations incomplètes. Enfin $MI(q_i, q_j|L, M_D)$ estime uniquement la dépendance qui apparaît dans le document. Cela se calcule selon :

$$MI(q_i, q_j|L, M_D) = \frac{N(q_i, q_j, R)N(*,*,R)}{N(q_i,*,R)N(*, q_j, R)}$$

L'article laisse planer une certaine ambigüité sur les caractéristiques de la structure *L*. Nous détaillons ce problème dans *(*Maisonnasse *et al.*, 2007*)*. Cette structure peut relier des positions dans la phrase ou relier des termes connus. Ce flou tend à donner des incertitudes sur le calcul des estimations. En effet, l'estimation de la probabilité $P(L|Q)$ soutient le fait que cette structure représente un ensemble de liens entre des mots connus (couples de mots). Or avec une telle interprétation, la décomposition à la base du modèle devient fausse, en effet *Q* se définit comme un ensemble de mots.

Si L représente un ensemble de couples de mots sur la requête, alors il contient les mots de la séquence Q et par conséquent :

$$P(Q|L, M_D) = 1$$

Les auteurs expérimentent ce modèle sur 6 collections de TREC sur lesquelles ils appliquent 48 requêtes. Ils testent leur modèle sur la moitié de cette collection après un réglage des différents paramètres de chaque modèle sur l'autre moitié. Les expérimentations montrent de meilleurs résultats pour ce modèle par rapport à ceux du modèle probabiliste ou ceux du modèle unigramme. Cependant la différence entre ce dernier et le modèle à base de dépendances reste faible, même si le modèle de dépendance donne de meilleurs résultats.

Les auteurs estiment que pour le moment, la prise en compte des dépendances entre les termes ne donne pas de bons résultats du fait de la difficulté d'estimer les dépendances entre les termes et de la difficulté à faire rentrer les dépendances et les termes dans un même modèle.

3 Conclusion

L'extension des modèles pour prendre en compte des informations de structure peut améliorer les résultats. Dès lors ces modèles sont à même de fournir de bons résultats car l'analyse syntaxique des phrases n'est pas utilisée pour la construction des termes, mais s'intègre directement dans le modèle. Ces méthodes proposent des modèles plus complets qui donnent un sens aux relations. Ils ont l'avantage de prendre en compte les spécificités de ces descripteurs.

Cependant la majorité de ces méthodes restent au niveau des mots et utilisent des analyses syntaxiques (ou statistiques). Elles sont fortement liées au fait que la structure utilisée représente un arbre, permettant des calculs des dépendances. Entre les deux modèles, le modèle de langue semble plus propice à l'intégration des relations autre que les dépendances. Il est facilement adaptable et fournit de bons résultats. C'est pour cette raison que nous l'avons choisi.

Les auteurs n'appliquent pas ces modèles sur des représentations sémantiques alors que c'est tout à fait envisageable. Dans la suite, nous verrons justement les travaux portant sur l'utilisation d'informations sémantiques. Nous détaillons certaines structures sémantiques et les modèles utilisés sur ces structures.

Chapitre V L'Information Sémantique

« Un concept est une invention à laquelle rien ne correspond exactement, mais à laquelle nombre de choses ressemblent. » Friedrich Nietzsche (Extrait de Posthumes)

Pour dépasser les limites dues à l'utilisation des informations syntaxiques, des travaux proposent l'utilisation d'informations sémantiques. Les méthodes les plus simples utilisent des descripteurs plus sémantiques en remplacement des mots-clefs, d'autres se basent sur des systèmes de représentation sémantique. Contrairement à l'information syntaxique où les travaux de traitement de la langue s'articulent autour de deux représentations des phrases, au niveau sémantique les travaux utilisent des structures plus diverses, par exemple le modèle des graphes conceptuels. Cette partie détaille donc ces différentes approches.

1 Descripteurs sémantiques

Utiliser des descripteurs sémantiques comme index semble une bonne idée car on s'affranchit alors des ambiguïtés du langage. On peut ainsi utiliser soit des **concepts**, soit des **acceptions** :

- Une **acception** est la signification d'un mot à partir de son usage dans la langue.
- Un **concept** représente une entité abstraite qui unifie et résume un ensemble d'objets concrets ou mentaux par abstraction de traits communs pertinents.

Des travaux étudient ces deux types de descripteurs sémantiques. Les deux sections suivantes présentent le détail de leur utilisation.

1.1 Acceptions

Des travaux de recherche d'information portent sur l'utilisation de sens notamment avec l'utilisation de base de données lexicales telles que WordNet[10] ou EuroWordNet (Gonzalo *et al.*, 1998)(Baziz, 2005). Ces bases de données fournissent un réseau d'acceptions dénotées sous le nom de *synsets,* ou *synsets* multilingues dans le cas d'EuroWordNet. De par leur nature, l'utilisation des acceptions nécessite l'usage de méthodes de désambiguïsation.

Dans (Voorhees, 1993), les auteurs déterminent le sens d'un mot en utilisant la distance sémantique entre chaque *synset* possible de ce mot et les *synsets* possibles des autres mots de la phrase. Les expérimentations sur plusieurs collections montrent que l'utilisation de *synsets* plutôt que de mots détériore les résultats.

(Gonzalo *et al.*, 1998) étudie l'impact de l'ambiguïté des termes à l'aide d'une désambiguïsation manuelle et l'introduction volontaire d'erreurs de désambiguïsation. Les auteurs montrent ainsi que le

[10] http://wordnet.princeton.edu/

système fonctionne mieux avec une indexation sémantique s'il fournit moins de 30% d'erreurs de désambiguïsation. Ces résultats se rapprochent de ceux de (Sanderson, 1994), où l'auteur montre que pour que la désambiguïsation apporte un plus pour la recherche d'information, cette désambiguïsation doit avoir une performance supérieure à 90%. Ces résultats soulignent la sensibilité des systèmes aux erreurs de désambiguïsation car celles-ci dégradent rapidement les résultats.

Dans sa thèse, Baziz (Baziz, 2005) propose une nouvelle technique de désambiguïsation et propose deux indexations basées sur les *synsets* de WordNet, l'une représentant un document comme un ensemble de *synsets,* l'autre comme un sous-arbre. Ces indexations donnent de moins bons résultats que les mots. Cependant une indexation basée sur une combinaison de *synsets* et de mots améliore la qualité contrairement à une indexation basée seulement sur des *synsets*. Les auteurs expliquent une partie de ces résultats par le trop faible recouvrement par la ressource (WordNet) sur le vocabulaire du corpus.

La principale difficulté dans l'utilisation des acceptions en recherche d'information concerne la désambiguïsation. Les expérimentations sont souvent menées sur des domaines généraux où les mots possèdent beaucoup de sens, ce qui rend encore plus difficile l'obtention de bons résultats. La construction de ressources, telles que WordNet avec une couverture exhaustive de la langue, s'avère une tâche difficile car la couverture de la langue n'est jamais obtenue, du fait de sa constante évolution et de l'existence de particularités lexicales dans chaque domaine de connaissance. Le résultat est souvent une couverture peu satisfaisante.

1.2 Concepts

Par nature, les concepts s'utilisent surtout dans des domaines restreints où des ressources permettent de décrire ces entités de manière plus ou moins formelle.

Par exemple, dans le domaine médical, des travaux utilisent le méta-thésaurus UMLS pour l'indexation de documents médicaux. Sur TREC genomics, Zou et al. (Zhou *et al.*, 2007) utilisent les termes en relation avec des concepts, par l'identification de ces termes dans la requête et dans le document. Ils améliorent légèrement les résultats des mots-clefs. Les auteurs effectuent ensuite des expériences en ajoutant des informations dans la requête. Ces expériences montrent que les meilleurs résultats s'obtiennent en ajoutant dans la requête l'ensemble des variations d'expression correspondant aux concepts reliés aux termes détectés dans la requête. Cette méthode d'extension de la requête se rapproche de l'utilisation directe des concepts.

Dans (Vintar *et al.*, 2003) ou (Aronson *et al.*, 1994), les auteurs évaluent directement des indexations conceptuelles. Les résultats obtenus dans le deuxième papier sont légèrement inférieurs à ceux obtenus par les mots-clefs. Récemment, dans la campagne d'évaluation CLEF médicale (2005-2007), nos expérimentations d'indexation à base de concepts ont montré de bons résultats : ils ont surpassé tous les autres types d'indexation. Cela témoigne de la pertinence de l'utilisation des concepts pour la recherche d'information, même si en fin de compte, l'amélioration des résultats n'est pas spectaculaire.

L'intérêt des concepts vient du fait que ce sont des abstractions de termes, contrairement aux acceptions qui représentent les sens d'un mot. Les concepts permettent ainsi de regrouper plusieurs termes sous un même identifiant ; ils permettent de factoriser différentes écritures d'un même concept. Les entreprises de conceptualisation se dédient souvent à un domaine unique. Se limiter à un domaine conduit à réduire le nombre potentiel d'ambiguïtés sur les termes du domaine. Il faut également noter que plus un terme est long, moins il a de chance d'être ambigu.

2 Logique terminologique

Une logique terminologique est un formalisme de représentation de connaissances. Elle contient trois composantes : le concept, le rôle et l'individu. Un ensemble de concepts et de rôles permet de construire des individus. Les concepts représentent des ensembles ou des classes. Un concept correspond à un ensemble de rôles qui expriment les relations existantes entre lui et d'autres concepts. Il existe deux types de concepts :

- Les **concepts primitifs** possèdent une description incomplète représentant un ensemble de conditions nécessaires à un individu pour appartenir à ce concept. Ces concepts s'introduisent par le symbole :<

CHAT :< (and ANIMAL (exactement 4 PATTE))

Un chat est un animal qui a quatre pattes, par contre tous les animaux à quatre pattes ne sont pas des chats.

- Les **concepts définis** possèdent une description complète des rôles nécessaires et suffisants à sa description. Ces concepts s'introduisent par le symbole :=

TRIANGLE := (and POLYGONE (exactement 3 COTE)

Tous les polygones qui ont trois côtés sont des triangles, la définition définit tous les éléments.

L'ensemble des concepts ainsi définis s'organise alors en une hiérarchie à l'aide de la relation de subsomption. Cette relation existe sous différentes formes : la définition extensionnelle A subsume B si l'ensemble des individus dénotés A contient l'ensemble des individus de B ; la définition intensionnelle A subsume B si tout individu décrit par B se décrit aussi par A .

Le modèle d'indexation de Sebastiani utilise cette représentation (Sebastiani, 1994) avec une extension à l'incertitude. Ce système appartient à la classe des modèles logiques. Chaque document est représenté comme un individu plutôt que par une expression logique. La requête se décrit quant à elle comme un concept. La pertinence d'un document pour une requête se détermine par l'implication $D \rightarrow Q$ correspondant alors à : *l'individu D correspond à une instance du concept Q* ou *la requête subsume le concept décrivant le document D.* Comme ce modèle ne permet pas d'ordonnancer par pertinence les documents retrouvés, (Meghini *et al.*, 1998) améliore le modèle en intégrant deux mesures de probabilités. Même si ce type de modèle est intéressant, son application reste difficile car elle nécessite d'une part une transformation des documents en expression logique, et d'autre part un système de correspondance basé sur un démonstrateur de théorème. L'efficacité en terme de vitesse de réponse n'est donc pas au rendez-vous.

3 Dépendance sémantique : RIME

Le système RIME (Recherche d'Informations MEdicales) (Berrut, 1988) propose un modèle de recherche d'information basé sur une représentation sémantique. Ce système est destiné à la recherche de rapports médicaux par des spécialistes du domaine. Ces spécialistes expriment des besoins d'informations précis, composés de concepts inter-reliés.

Le système RIME est adapté à des utilisateurs qui ont besoin de décrire leur besoin d'information de manière précise et complète. Le système fournit en retour des résultats aussi précis que le sont les requêtes. Il s'agit d'un système orienté précision basé sur un langage complexe ; ce système est adapté au domaine de la radiologie et à la nature intrinsèque des documents recherchés. En effet, les

radiologues utilisent une langue fortement technique et formelle. Cela rend possible une interprétation plus complète et moins ambiguë des documents. Les auteurs estiment néanmoins que le modèle proposé reste transposable à d'autres domaines ayant les mêmes caractéristiques.

3.1 Le modèle des documents de RIME

Ce modèle représente chaque phrase par une structure similaire à un arbre, basé sur des dépendances sémantiques. Dans cette représentation, les nœuds feuilles représentent des concepts primitifs du domaine et les nœuds non terminaux représentent des opérateurs sémantiques. Les opérateurs sémantiques décrivent des opérateurs binaires qui explicitent le lien sémantique entre deux sous-arbres, représentant eux-mêmes des concepts.

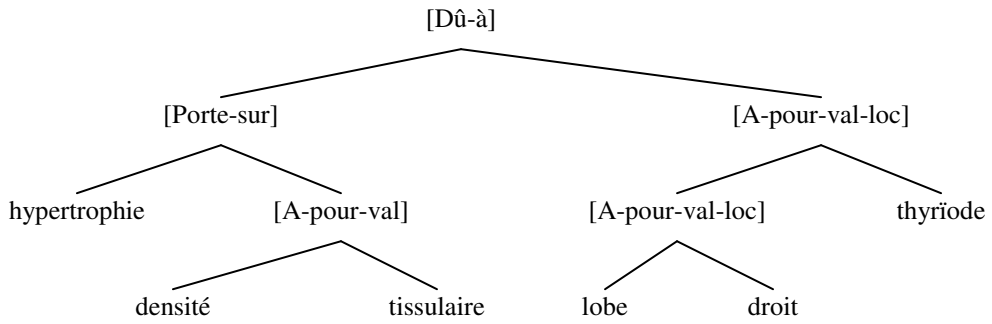


Figure 21 Arbre de la phrase 'hypertrophie de densité tissulaire du lobe de la thyroïde'

L'approche RIME (Berrut, 1988) considère deux types de concepts : d'une part les concepts primitifs qui correspondent aux feuilles de l'arbre. Ces concepts correspondent à des faits médicaux ou à des termes techniques qui constituent le vocabulaire du domaine ; d'autre part, les concepts complexes représentés par des sous-arbres et constitués d'au moins un opérateur sémantique.

De manière générale, un concept est constitué d'un ensemble de concepts plus fins structurés par des relations sémantiques. Par exemple *'hépatomégalie'* se décompose en *'hypertrophie du foie'*, hypertrophie se décomposant lui-même en *'augmentation du volume'*. Le vocabulaire de base correspond aux faits élémentaires définis par le modèle. Il constitue la limite de la décomposition.

L'ensemble des arbres construits dans RIME respecte un modèle formel défini à l'aide d'une grammaire. Tous les arbres manipulés résultent d'une phase de compréhension traduisant les textes médicaux en structure.

3.2 Modèle d'interrogation

L'évaluation de la requête dans le modèle RIME se base sur le modèle logique. Ce modèle évalue la plausibilité d'une implication logique entre la requête Q et le document D dans un certain système de connaissance K ; $P_K(D \rightarrow Q)$. Si cette implication ne peut pas s'obtenir, le système doit alors évaluer l'apport d'information nécessaire à la modification de l'un des trois paramètres (Q, D, K) pour que l'implication devienne possible. Cette méthode correspond au principe d'incertitude de Nie (Nie, 1990), qui formule la mesure de l'incertitude de l'implication d'une proposition x par une proposition y. Cette incertitude correspond à l'extension minimale à apporter à l'ensemble d'information pour établir la vérité de $y \rightarrow x$.

Dans RIME, cette correspondance entre la requête et le document se base sur un nombre restreint de règles de dérivation qui exploitent des informations sémantiques pour établir la plausibilité de la requête.

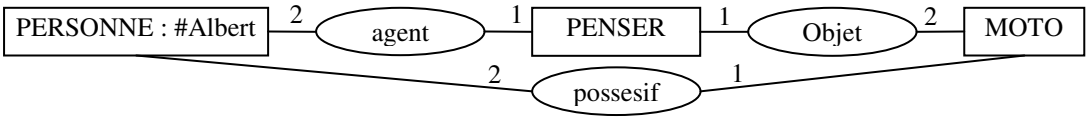
3.3 Limites du modèle

Une limite de ce modèle vient de l'établissement du vocabulaire de base. Le système doit définir explicitement ce vocabulaire par la construction d'un dictionnaire sémantique, c'est-à-dire une structure qui associe à chaque terme un arbre sémantique. Dans le modèle RIME, ce dictionnaire a été construit avec les praticiens. Cette phase est importante, et comme le souligne Berrut page 72 de sa thèse (Berrut, 1988), le vocabulaire de base fixe le niveau de compréhension de tout le système car tout le reste du vocabulaire du système s'exprime à partir de celui-ci. Une autre difficulté réside dans la construction de toutes les règles qui permettent la construction des représentations des phrases mais aussi dans la définition des règles de dérivation permettant la correspondance.

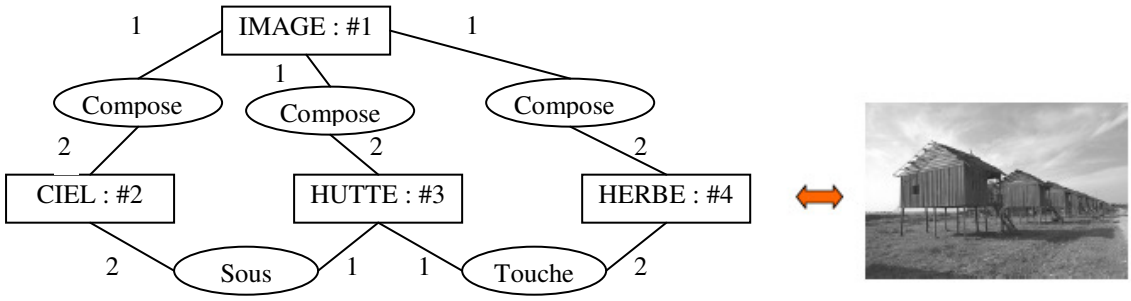
4 Graphe conceptuel

4.1 Le formalisme

Sowa à la fin des années 70 (Sowa, 1984) introduit le modèle des graphes conceptuels. Ces graphes permettent la représentation des connaissances d'un domaine, en particulier à celles contenues dans des énoncés. Il s'appuie sur l'étude de la perception en psychologie. Le but du modèle consiste à fournir un formalisme d'expression de la sémantique qui soit logique, précis, facilement lisible par les êtres humains et d'une complexité suffisamment raisonnable pour que des systèmes informatiques puissent l'utiliser. La figure 22 présente deux exemples de graphes conceptuels.



Exemple 1 : pour le texte '*Albert pense à sa moto*' (Mugnier et Chein, 1996)



Exemple 2 : graphe correspondant à l'image de droite : l'image est composée d'une hutte sous le ciel et qui touche l'herbe

Figure 22 Deux exemples de graphes conceptuels

Nous présentons par la suite un résumé de la formalisation du modèle de base de Sowa proposée par Chein et Mugnier (Genest et Chein, 2005).

4.1.1 Support des graphes conceptuels

Le vocabulaire utilisé dans les graphes se définit par un support. Celui-ci comprend deux treillis : l'un pour les concepts et l'autre pour les relations. Ces treillis organisent le vocabulaire à l'aide de relations de type *'sorte de'* qui forment deux ordres partiellement ordonnés notés '≥'. Le treillis des concepts $T_{concepts}$ représente l'ensemble des types de concepts. Cet ensemble partiellement ordonné comprend par définition un plus grand élément '⊥', le type absolu, et un plus petit élément '⊤', le type absurde. La figure 23 présente une vue partielle d'un treillis de types de concepts.

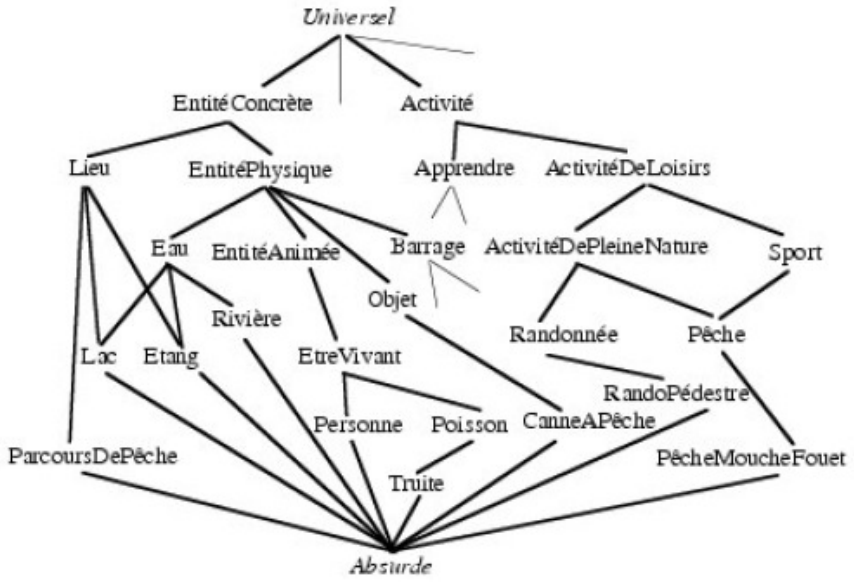


*Figure 23 Vue partielle d'un ensemble de type (*Mugnier et Chein, 1996*)*

De ce fait, tout couple de types de concepts possède un sur-type commun minimal et un sous-type commun maximal. Le treillis des relations $T_{relations}$ représente l'ensemble des types de relations. Un ordre partiel régit chaque sous-ensemble de relations de même arité, la Figure 24 représente une vue partielle d'un de ces treillis.

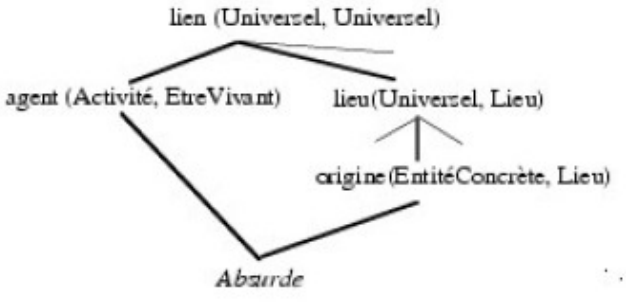


*Figure 24 Vue partielle de l'ensemble des relations binaires (*Mugnier et Chein, 1996*)*

À ces deux treillis s'ajoute l'ensemble *I* des marqueurs individuels qui servent de référents vers des éléments nommés dans les graphes. Un marqueur générique s'ajoute à cet ensemble, noté *, il désigne l'ensemble des éléments d'un type.

Le rôle du support est de fixer la manière dont s'expriment les connaissances.

4.1.2 Format des graphes conceptuels

Un graphe conceptuel *G* représente un graphe biparti, c'est-à-dire ayant deux types de nœuds. Il se définit par *G=(C, R, E, Étiquette)* où :

- *C* représente l'ensemble des nœuds concept (représentés par des rectangles).
- *R* représente l'ensemble des nœuds relation (représentés par des ovales).
- *E* représente l'ensemble d'arêtes reliant un nœud concept à un nœud relation. Les arêtes adjacentes à un nœud relation possèdent un ordre, celui-ci se représentant par une numérotation des arêtes.
- *Étiquette* représente la fonction de correspondance qui donne à chaque nœud une étiquette.

Un sommet concept se représente alors par la paire [type(c) : ref(c)] où *ref(c)* le référent du concept appartient à *I* et où *type(c)* le type du référent appartient à T_C (noté en majuscule) [PERSONNE : Paul] ou [SKY : #2]. Une relation de conformité doit relier le type et le référent. Si le sommet possède un référent générique alors ce concept correspond à concept générique par exemple [PERSONNE] ou [PERSONNE:*] ; sinon le concept représente un sommet individuel : [PERSONNE : Paul].

Un sommet relation se représente seulement par le type de relation *(type(r))* appartenant à T_R. Chaque concept relié par la relation doit avoir un type égal ou inférieur au type de l'argument défini dans le treillis du type de relation auquel il correspond.

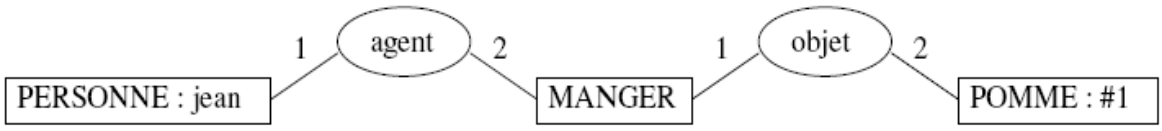


Figure 25 Graphe conceptuel représentant 'jean mange une pomme'

Pour simplifier nous pourrons par la suite représenter un graphe conceptuel par *G(C,R)* tel que défini par Sowa, où *C* représente l'ensemble des concepts contenant le nom et le référent et *R* l'ensemble des relations constituées du nom de relation et de ses liens avec les concepts.

4.1.3 La projection

L'opération de projection permet de mettre en place un pré-ordre sur les graphes : soit *G* et *H* deux graphes, si *H* se projette dans G alors $G \geq H$. On dit aussi que H constitue une généralisation de *G* si *H* s'obtient à partir d'un nombre fini d'opérations de généralisation sur *G* et inversement que *G* constitue une spécialisation de *H*. La projection existe s'il existe à l'intérieur d'un graphe des sous-graphes de même structure qu'un graphe donné dont les arcs et les nœuds peuvent se restreindre.

Définition : la projection d'un graphe conceptuel *H* vers un graphe *G* correspond à une paire d'applications de *H* vers *G*, l'une sur les concepts et l'autre sur les relations des deux graphes, telles que :

- Elles conservent les arêtes et leur numérotation.
- Elles peuvent restreindre les étiquettes des sommets relation ou concept.

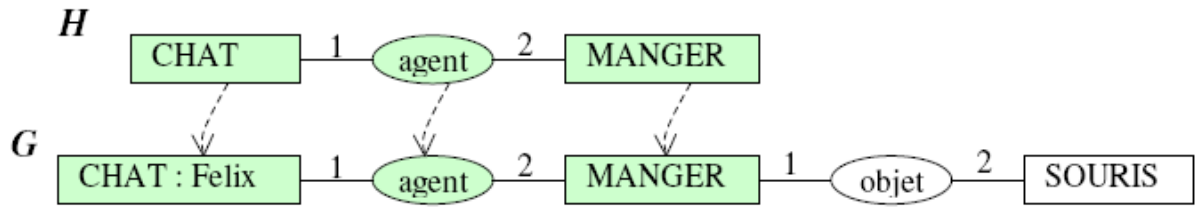


Figure 26 Exemple de projection de H dans G

4.2 Les Graphes conceptuels en recherche d'information

Dans le formalisme des graphes conceptuels, l'opération de projection permet de trouver des composantes spécialisées d'un graphe au sein d'un autre, c'est-à-dire qu'elle permet de savoir si un graphe possède une partie sémantiquement équivalente (ou plus précise) qu'un autre graphe. Cette opération de projection s'utilise dans les systèmes de recherche d'information pour faire la correspondance entre le graphe conceptuel d'un document et celui d'une requête. Si une requête se projette dans un document, alors le document répond avec pertinence à la requête.

4.2.1 Un modèle logique

Des travaux montrent que l'opération de projection équivaut à l'implication en logique du premier ordre. Si un graphe *H* se projette dans un graphe *G*, la formule logique de *G* implique celle de *H*. Or la pertinence d'un document *D* vis-à-vis d'une requête *Q* s'exprime dans le modèle logique par l'évaluation de l'implication $D \rightarrow Q$. La pertinence d'un document représenté par un graphe conceptuel *D* pour une requête représentée par un graphe *Q* revient donc à vérifier l'existence d'une projection de *Q* dans *D*. Les modèles de recherche d'information à base de graphes s'appuient donc sur le modèle logique.

Cependant, l'opération de projection ne satisfait pas totalement ce modèle. En effet un système de recherche d'information doit permettre de classer les réponses par ordre de pertinence, or la projection fournit un résultat booléen qui ne permet pas d'ordonner les documents selon un ordre de pertinence. L'utilisation de la projection peut, de plus, donner des résultats considérés comme trop précis, le document devant satisfaire totalement la requête. Cette condition élimine les documents satisfaisant partiellement la requête. Dans le modèle logique de recherche d'information, ces problèmes se posent vis-à-vis de l'application du principe d'incertitude. Ce principe s'applique sur l'implication de la requête par le document, la mesure de cette incertitude permettant de pondérer les résultats. Or la projection ne répond pas à l'application de ce principe en recherche d'information du fait qu'elle donne une certitude comme résultat.

4.2.2 Amélioration du modèle

Les travaux qui utilisent les graphes conceptuels proposent plusieurs modifications du modèle des graphes conceptuels de manière à rendre la projection plus utilisable en recherche d'information.

4.2.2.1 La projection partielle

Chevallet propose dans (Chevallet, 1992) l'utilisation d'une projection partielle basée sur les notions de projection compatible et de généralisation commune dans le but de rendre l'interrogation des graphes compatible avec le principe d'incertitude. Il existe une généralisation commune entre deux graphes *G* et *G'* s'il existe un troisième graphe *H* tel que ce graphe soit une généralisation de *G* et de *G'* ($G \leq H$ et $G' \leq H$). La projection partielle repose sur la possibilité que seul un sous-graphe de la requête se projette sur le document.

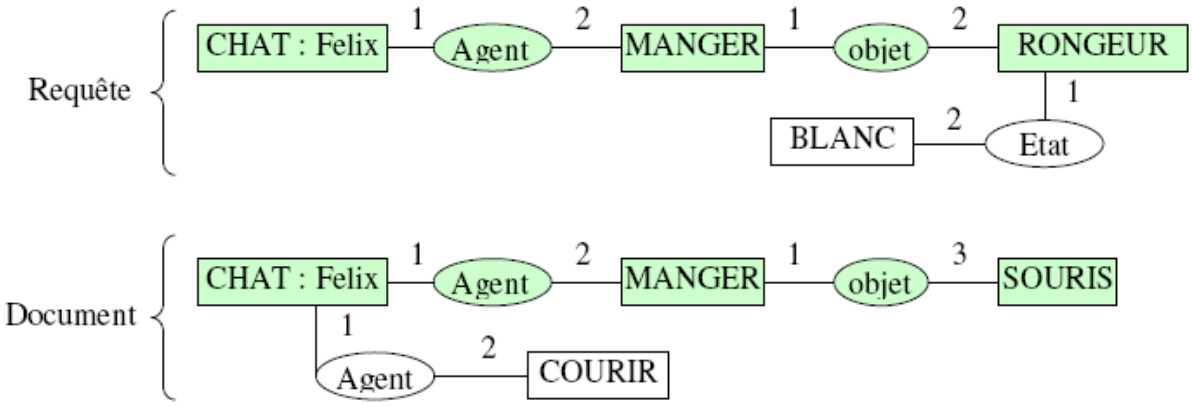


Figure 27 Exemple de projection partielle

Dans l'exemple de la figure 27, le graphe de la requête pris entièrement ne se projette pas dans le document, pourtant le document peut correspondre à la requête à la différence que la couleur de la souris n'est pas connue. Il existe un sous-graphe dans la requête qui se projette dans le document, il s'agit une projection partielle. La projection partielle de *H* sur *G* consiste donc à trouver une bijection entre un sous-graphe connexe de *H* et un sous-graphe connexe de *G* de manière à ce que les graphes aient la même structure et que les concepts ou les relations en bijection aient un sous-type en commun.

4.2.2.2 Projections à une transformation près

Pour sa part Genest (Genest, 2000) propose de rechercher des projections à une transformation près. Il définit le principe d'incertitude sur les graphes comme la transformation minimale du graphe *D* du document en un graphe *D'* tel que le graphe *R* de la requête se projette dans *D*'. Pour cela il définit la notion de séquence de transformation comme un ensemble de transformations élémentaires pouvant s'appliquer à un graphe. Il sélectionne ensuite un sous-ensemble de ces séquences, dites acceptables, tel que la modification du sens apporté à un document par ces séquences soit jugée assez proche du sens initial du graphe.

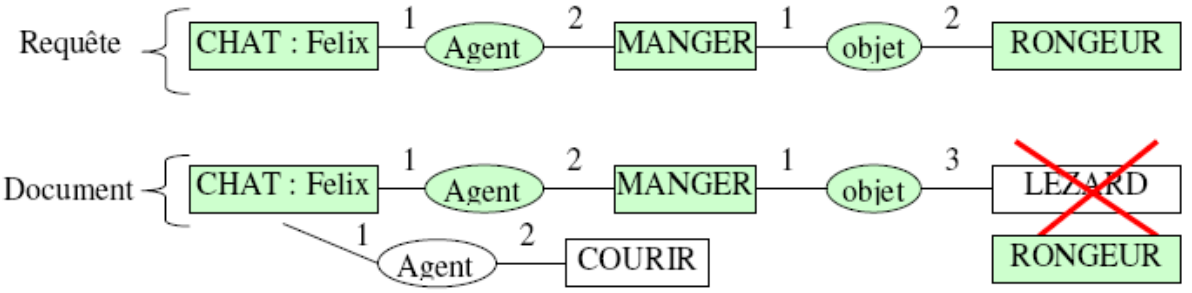


Figure 28 Exemple de projection à une transformation prés

Dans l'exemple de la figure 28 la requête ne se projette pas dans le document ; par contre si on modifie l'étiquette LEZARD du document par l'étiquette RONGEUR alors la projection existe. Il existe donc une projection à une transformation près. La séquence de transformation est constituée d'une seule transformation : le changement d'étiquette d'un sommet. L'auteur munit l'ensemble de ces séquences d'un pré-ordre total afin de les ordonner, la séquence vide représentant le plus petit élément de cet ensemble. Ainsi le système peut ordonner les projections à une transformation près par rapport à

l'ordre des séquences qu'elles utilisent et par conséquent ordonner les résultats d'une requête. L'utilisation de cette méthode augmente le nombre de projections à tester. Pour un document, le système doit tester la projection de la requête sur toutes les transformations obtenues à l'aide de l'ensemble des séquences acceptables, ce qui reste complexe.

L'auteur présente une application basée sur une ontologie. Dans ce système, l'utilisateur guide l'indexation. Une première étape consiste en l'analyse du texte et en l'extraction des éléments de l'ontologie qui apparaissent dans le texte. Dans une seconde étape, l'utilisateur a la possibilité de créer les relations entre ces éléments. L'utilisateur crée les requêtes à l'aide d'un outil de construction de graphes. La correspondance s'effectue finalement à l'aide de la projection à une transformation près.

4.2.3 Implémentation des modèles de recherche d'information basés sur les graphes conceptuels

Si certains travaux tentent de résoudre le problème de l'utilisabilité de la projection en recherche d'information, d'autres visent à réduire sa complexité. L'opération de projection correspond à un calcul NP-complet (Mugnier et Chein, 1996), ce qui rend son application difficile sur de grands corpus, tout en maintenant un temps de réponse raisonnable. Des travaux portent donc sur la réduction de cette complexité.

4.2.3.1 Projection étendue

Le modèle des graphes sémantiques présenté dans (Roussey, 2001) propose d'améliorer la description sémantique et multilingue des documents par une instanciation du modèle des graphes conceptuels dans le but d'obtenir un formalisme plus proche des langages documentaires. Ce modèle constitue une première tentative de création d'un langage de représentation multilingue à base de graphes.

Pour améliorer la projection, l'auteur met en place un opérateur de projection étendue ; cet opérateur ne prend pas en compte la structure du graphe. La projection étendue se forme d'un couple de fonctions, l'une sur les concepts, l'autre sur les relations des deux graphes. Cet opérateur a les propriétés suivantes :

- Il ne conserve pas les sommets concepts.
- Il conserve les arcs.
- Il peut restreindre ou augmenter le type des concepts et les arcs.

La projection étendue ne tient pas compte de la globalité du graphe. Pour résoudre ce problème, les graphes utilisés dans (Roussey, 2001) sont considérés sous forme normale : un nœud ne peut pas apparaître deux fois dans le graphe, par conséquent pour un ensemble de nœuds et d'arcs, il n'existe qu'une seule structure de graphes possible. L'auteur suppose que cette réduction n'a que peu d'impact sur la recherche d'information car habituellement une notion apparaît de manière unique dans un document.

4.2.3.2 Table d'accélération

Ounis (Ounis et Pasca, 1998) s'oriente vers un système à base de fichiers inverses pour résoudre le problème de la complexité de la projection. L'idée de cette approche consiste à effectuer une partie des calculs complexes de la projection au moment de l'indexation de manière à alléger ceux effectués au moment de l'interrogation. En recherche d'information, c'est principalement la rapidité de l'opération d'interrogation qui compte. Pour créer le fichier inverse, l'auteur décompose la structure des graphes en *relons* et caractérise chaque nœud du graphe de manière unique à l'aide de *témoin*.

Un relon se présente à l'aide du type de relation et de la suite de ses arguments : <*type de relation, argument1,… argument n*>. Ces relons s'obtiennent à l'aide de l'opération d'éclatement du graphe (Amati et Ounis, 2000).

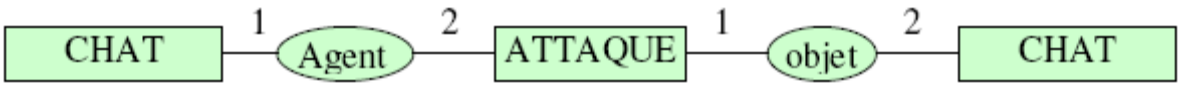


Figure 29 Graphe conceptuel représentant la phrase : 'un chat attaque un chat'

Le graphe de la figure 29 se décompose en deux relons <Agent, CHAT.1, ATTAQUE> et <objet, ATTAQUE, CHAT.2>. Dans ce cas, l'utilisation de témoins permet d'apprendre qu'il peut exister deux chats différents ou non. Le système peut reconstruire sans ambiguïté le graphe à partir de ces relons.

Le système utilise ensuite ces relons pour construire le fichier inverse associant à chaque relon la liste des documents où le relon apparaît. À cela s'ajoutent des tables d'accélérations pour chaque type de relation. Ces tables indiquent, pour chaque type de concept d'une relation, les relons du fichier inverse qui spécialisent ce concept. Les tables d'accélération permettent de pré-calculer les spécialisations des relons.

La fonction de correspondance s'effectue en recherchant dans la table d'accélération les correspondances aux relations présentes dans le graphe de la requête. A l'aide de ces correspondances dans le fichier inverse, le système obtient la liste des fichiers contenant les relations de la requête ou leurs spécialisations. Si tous les relons d'une requête correspondent à un relon égal ou spécialisé dans un document alors la requête se projette dans ce document.

4.2.3.3 Graphe étoile

Une autre méthode pour indexer les graphes conceptuels consiste à ne pas prendre en compte le graphe dans sa globalité mais partiellement. Pour cela le graphe initial est décomposé en graphes étoiles. Un graphe étoile forme un graphe qui ne contient qu'une seule relation. Les graphes étoiles s'obtiennent en éclatant le graphe initial en pièces élémentaires (Amati et Ounis, 2000).

Les sous-graphes ainsi extraits représentent le document, ils peuvent donc s'utiliser comme les descripteurs de celui-ci. Ces descripteurs peuvent s'utiliser en tant que dimension d'un vecteur pour représenter un document. La mise en place d'un modèle vectoriel permet à la fonction de correspondance de retrouver des documents qui satisfont partiellement la requête et de classer ces documents selon leur pertinence. Pour cela, l'auteur calcule le poids des différentes dimensions pour chaque document. Les graphes étoiles s'utilisent essentiellement dans l'indexation d'images (Martinet, 2004).

Pour prendre en compte l'existence du treillis des concepts, le système étend les requêtes de manière à contenir toutes les spécialisations des graphes étoiles présents dans la requête initiale. L'opération ainsi exécutée se rapproche de celle de la projection mais en moins complexe.

5 Conclusion

Cette partie relate différentes approches de l'utilisation des représentations sémantiques en recherche d'information. Cependant, l'indexation au niveau sémantique des documents se révèle difficile. La précision des représentations et leur caractère complexe rendent très difficile la mise en

place d'une indexation automatique. La création de ces index ne peut alors s'appliquer qu'à des corpus de documents restreints.

Nous émettons deux remarques sur ces travaux. Premièrement nous notons que lorsque les travaux proposent une extraction automatique des documents, ces travaux utilisent pour la plupart des informations syntaxiques qui servent de base pour la création des représentations sémantiques. Si l'utilisation de l'information syntaxique n'apporte pas d'amélioration, cette information reste cependant nécessaire pour construire des représentations sémantiques comme postulé dans la théorie Sens Texte. Deuxièmement nous remarquons la prédominance du modèle logique de recherche d'information lors de l'utilisation d'indexations basées sur des structures sémantiques, alors que ce modèle reste peu utilisé sur les mots-clefs car d'autres modèles fournissent de meilleurs résultats.

Nous pensons que l'amélioration de la précision des systèmes de recherche d'information passe par l'utilisation de méthodes traitant l'information sémantique contenue dans les documents mais qui ne nécessitent ni des méthodes complexes ni la création de ressources.

Actuellement, la normalisation des langages de description de concepts encourage la construction de ressources sémantiques (ex: OWL pour web sémantique), tandis que des efforts de fusion de ressources spécialisées permet une large couverture de la langue (ex:UMLS). Ces ressources consistent en des ontologies ou des thésaurus, qui la plupart du temps n'aboutissent qu'à la mise en place d'indexations par sacs de concepts. Peu de travaux portent sur l'utilisation de ces ressources pour créer des représentations structurées des documents, encore moins pour développer sur ces structures des modèles autres que le modèle vectoriel ou le modèle logique, c'est ce que nous proposons de faire dans cette thèse.

Bilan

Dans cet état de l'art nous avons exploré différents travaux sur l'intégration d'informations, notamment linguistiques, en parcourant l'axe de l'expressivité. Nous nous sommes plus particulièrement intéressés aux informations structurelles avec l'utilisation de structures de dépendance ou encore de graphes conceptuels. Sur cet axe, l'état de l'art montre l'intérêt de l'utilisation de représentations expressives. L'utilisation de telles représentations reste conditionnée par la disponibilité de traitements de la langue permettant leur création. L'état de l'art montre également que les modèles basés sur des structures, induites par l'utilisation de relations, permettent d'améliorer la précision des premières réponses du système. Ces différentes approches se positionnent ou donnent des informations sur le niveau d'expressivité qu'elles utilisent. Cependant, elles ne modélisent pas explicitement cette expressivité. Par conséquent, comparer l'expressivité de deux modèles se révèle difficile. Nous proposons par la suite l'utilisation de supports de vocabulaires pour définir des modèles. Les supports de vocabulaires mettent en évidence l'expressivité du modèle et permettent la comparaison des modèles sur cette expressivité.

Au final, sur le texte, de nombreux systèmes possèdent une expressivité faible et se basent uniquement sur des mots-clefs. Cette expressivité est plus fortement mise en avant dans les systèmes de recherche d'information vidéo ou image. Dans ces systèmes, les documents se représentent souvent à l'aide de plusieurs points de vue, par exemple couleur et ou texture pour les images.

Par la suite nous présentons un cadre permettant de modéliser des systèmes de recherche d'information par l'utilisation d'un support de vocabulaires. Nous décomposons l'expressivité en deux parties, d'une part les points de vue sur le document, et d'autre part le niveau d'expressivité de ces points de vue.

L'utilisation d'une modélisation à base de supports de vocabulaires met aussi en avant des critères supplémentaires sur l'utilisation des supports. Nous sélectionnons deux critères correspondant à l'utilisation du support à l'indexation. Le premier caractérise ces modèles en fonction de la portée des vocabulaires du support, c'est-à-dire l'ensemble des éléments constituant le support et potentiellement utilisables par le modèle. Le second les caractérise en fonction de la portée de la représentation des documents sur ce support, c'est-à-dire la portion du support de vocabulaires utilisée pour représenter un document.

Si les approches présentées dans l'état de l'art se positionnent par rapport à l'expressivité, elles ne se positionnent pas ou très peu par rapport à la taille et à l'utilisation de ce support à l'indexation. Sur la figure 30 nous positionnons quelques systèmes présentés dans l'état de l'art sur le plan de l'utilisation des supports de vocabulaires : les graphes conceptuels possèdent un support exhaustif mais ne représentent que le contenu du document. Les modèles de langue sur les dépendances établissent pour un document toutes les probabilités des éléments du support mais limitent ce support à la collection. Les modèles vectoriels sur les mots-clefs, les syntagmes ou encore les concepts, ne modélisent que le vocabulaire de la collection et sur ce vocabulaire ne représente que le contenu du document.

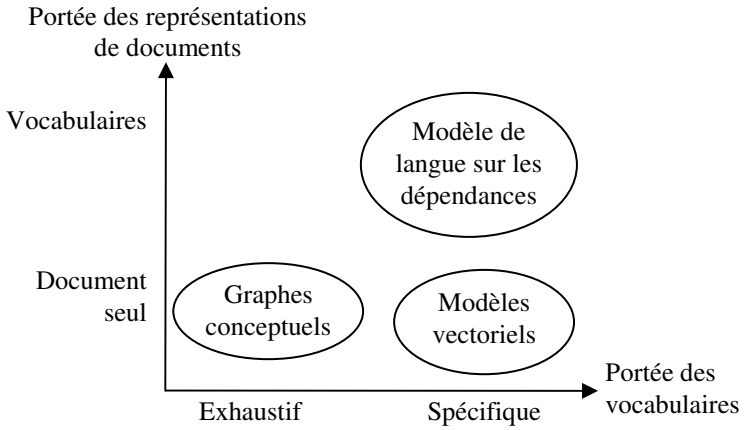


Figure 30 Positionnement de certains modèles dans l'espace formé par l'utilisation des supports de vocabulaires pour l'indexation

Dans la suite, nous détaillons un cadre permettant de définir des modèles de recherche d'information faisant explicitement apparaître la notion de supports de vocabulaires. Une fois ce cadre défini, nous proposons deux modèles expressifs. Ces deux modèles se positionnent au même niveau sur l'axe de l'expressivité des supports, mais s'opposent sur le plan de l'utilisation des vocabulaires à l'indexation.

PARTIE 3 : MODELISATION DE L'EXPRESSIVITE

Introduction

Nous proposons des modèles de recherche d'information orientée précision. Avant de présenter ces modèles, nous définissons dans un premier temps un cadre pour la formulation de modèles expressifs. Ce cadre met en avant la construction de *supports de vocabulaires* qui permettent de modéliser la représentation du document et celle des requêtes. Ces supports de vocabulaires manipulent des vocabulaires constitués de *types*, définis à l'aide d'un *support de type*. Enfin, ce cadre détaille la correspondance entre une représentation de document basée sur le *support de vocabulaires des documents* et une représentation de requête basée sur le *support de vocabulaires des requêtes*.

La définition du support de vocabulaires dans un modèle de recherche d'information détermine l'expressivité de ce modèle. Cette expressivité se définit en premier lieu par le nombre de vocabulaires utilisés qui dénote le nombre de points de vue représentant le document. Elle se définit en second lieu par les *types* qui forment les unités de ces vocabulaires et qui dénotent le niveau d'expression d'un point de vue. Elle s'exprime enfin par le nombre de *types* utilisés par un vocabulaire qui définit la complexité d'expression d'un point de vue.

Nous proposons de construire deux modèles orientés précision à partir de ce cadre. Nous choisissons deux modèles d'expressivité similaire qui utilisent des représentations sémantiques à base de graphes. Ces représentations utilisent plusieurs points de vue : conceptuel et relationnel qui sont des points de vue sémantiques.

Indépendamment de leur expressivité, nous positionnons ces deux modèles sur des approches différentes de l'utilisation du support de vocabulaires lors de l'indexation. Les deux modèles s'opposent diamétralement sur la portée des vocabulaires du support et sur la portée des représentations de documents. Cela nous permet de comparer deux approches différentes de l'utilisation des supports de vocabulaires.

Le premier modèle, le modèle local, se base sur des vocabulaires exhaustifs et ne représente que le contenu du document. Ce modèle s'inspire des modèles des graphes conceptuels pour représenter les documents et les requêtes. Il utilise une fonction de correspondance basée sur la projection pour calculer la pertinence d'un document.

Le deuxième modèle, le modèle global, se base sur des vocabulaires spécifiques qui se limitent aux éléments de la collection. Sur ces vocabulaires, les documents sont représentés à l'aide de la totalité des vocabulaires. Ce modèle est un modèle original qui s'inspire des modèles de langue et propose des modèles de graphes. Ce modèle évalue la probabilité de génération d'un graphe de requête en fonction du modèle de graphe d'un document.

Nous présentons donc un premier chapitre qui détaille le cadre utilisé pour décrire les deux modèles. Ce cadre définit des supports de vocabulaires qui modélisent l'expressivité des représentations. Dans un deuxième chapitre nous utilisons ce cadre pour détailler deux modèles orientés précision.

Chapitre VI Cadre Général des Modèles

« Il n'y a pas de savant qui ne pense continuellement par modèles – même s'il ne se l'avoue ni aux autres ni a lui-même » Auger 1965 (les modèles dans la science)

Nous présentons dans cette partie un cadre général permettant de définir des modèles de recherche d'information. Ce cadre met plus particulièrement en avant l'expressivité des modèles. Un modèle se divise en trois :

- un **modèle de document**, qui détermine la représentation du document dans l'index,
- un **modèle de requête**, qui détermine la représentation d'une requête,
- un **modèle de correspondance** entre la requête et les documents, qui établit la pertinence entre la représentation d'une requête et celle d'un document.

Ces modèles se basent sur des éléments concrets :

- un **corpus de documents**, déterminant l'ensemble des documents accessibles,
- des **besoins d'informations**, qui représentent des lacunes de connaissance qu'un utilisateur souhaite combler (Belkin, 1980).

Et il propose des solutions à l'aide des éléments suivants :

- un **support de types** qui contient les *briques de base* nécessaires à la représentation des documents et des requêtes,
- des **supports de vocabulaires** qui permettent de définir le modèle de document et le modèle de requête. Ces supports fixent l'expressivité des représentations des documents et des requêtes,
- un **corpus indexé**, constitué de l'ensemble des représentations de documents, représenté selon le modèle de document,
- une **fonction d'indexation** qui, à partir d'un document brut, traduit ce document dans le modèle de document,
- des **requêtes** établies selon le modèle de requête,
- une **fonction de composition de la requête**, qui aide l'utilisateur à traduire son besoin en information dans le modèle de requête.

Le modèle de recherche d'information détermine la représentation des documents du corpus et des requêtes manipulées par un modèle de document et un modèle de requête ainsi que leurs liens de pertinence. Ce modèle traduit les besoins de recherche d'information en un système permettant leur résolution comme le montre la figure 31.

Dans ce chapitre, après avoir introduit les éléments manipulés par les modèles, nous présentons un cadre général pour modéliser les systèmes de recherche d'information.

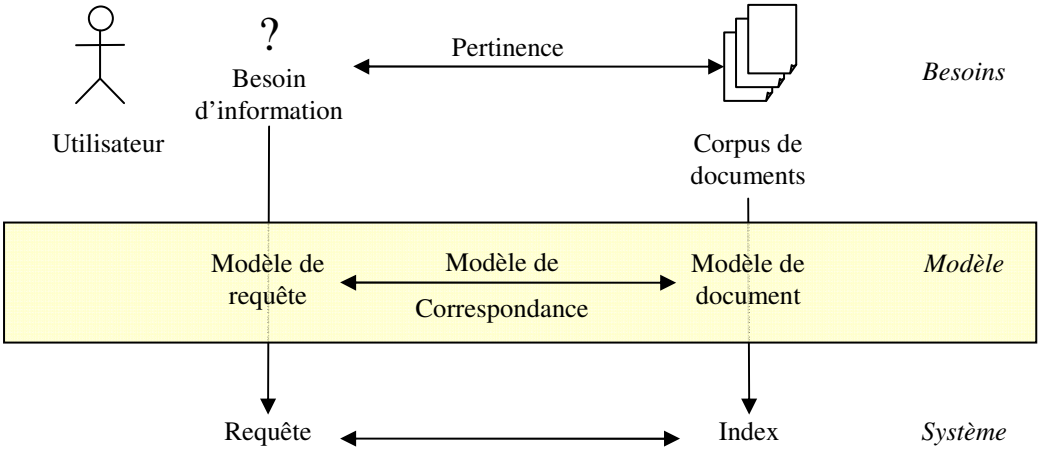


Figure 31 Positionnement du modèle dans la traduction des besoins en un système de recherche d'information

1 Définitions et supports

Nous définissons au préalable plusieurs éléments qui servent à la formation d'un modèle de recherche d'information.

1.1 Support de types d'un système de recherche d'information

Le support de types définit les types de base manipulés par un système de recherche d'information.

1.1.1 Les types d'un système de recherche d'information

Nous disposons d'un certain nombre de types *T*. Un T_{type} se compose d'éléments *t* ayant des caractéristiques communes, par exemple :

- T_{mots} un ensemble de mots,

  Exemple :

$$T_{mots}=\{les, variétés, pomme, de, terre, \ldots\}$$

  Ce type définit l'ensemble des mots pouvant être détectés dans des textes, sans filtrage.

- T_{termes} un ensemble de termes,

  Exemple :

$$T_{termes}=\{variétés, pomme\ de\ terre, \ldots\}$$

  Ce type définit l'ensemble des termes pouvant être détectés dans des textes, il ne contient pas les mots outils et contient des mots composés.

- $T_{concepts}$ un ensemble de concepts comme ceux définis dans UMLS[11],

  Exemple :

$T_{concepts}$={C0817096(poumon), C0032225(plèvre), C0018787(cœur), C0222762(cage thoracique), …}

  Ce type définit un ensemble de concepts. Un concept est une entité abstraite que nous écrivons à l'aide d'un identifiant (ex : *C0817096*) et d'un terme explicitant le concept indiqué entre parenthèses (ex : *(poumon)*). Pour plus de détails la section 1.1 du Chapitre IX , donne les détails de la définition d'un tel type à l'aide d'UMLS.

- $T_{synsets}$ un ensemble de *synsets* appartenant à WordNet[12],
- $T_{relations}$ un ensemble de noms de relations comme ceux définis par UMLS,

  Exemple :

$T_{relations}$={localisation, mesure, est partie de, touche…}

  Ce type définit l'ensemble des noms de relations sémantiques qui peuvent être détectées entre deux concepts.

1.1.2 Support de types

a) Définition

On appelle support de types *ST* la liste des types utilisés par un système de recherche d'information. Un support de types *ST* est constitué d'un n-uplet formé de *nst* types que l'on ordonne de 1 à *nst*.

$ST=(T_1, T_2, \ldots, T_{nst})$ avec $nst \geq 1$

b) Exemple

Nous donnons ci-dessous deux exemples de supports de types, l'un pour un système basé sur les mots-clefs, l'autre pour un système basé sur des graphes, nous illustrons ces deux exemples graphiquement par la figure 32 :

$ST_{mots} = (T_1)$	$ST_{graphes} = (T_1, T_2)$
$nst=1$	$nst=2$
$T_1=T_{mots}$	$T_1=T_{concepts}$, $T_2=T_{relations}$

Ces deux supports s'écrivent aussi sous la forme :

ST_{mots}=({les, variétés, pomme, de, terre, …})

$ST_{graphes}$ =({C0817096(poumon), C0018787 (cœur), C0222762(cage thoracique), …},

{localisation, mesure, est partie de, touche…})

[11] http://umlsinfo.nlm.nih.gov

12 http:// wordnet.princeton.edu

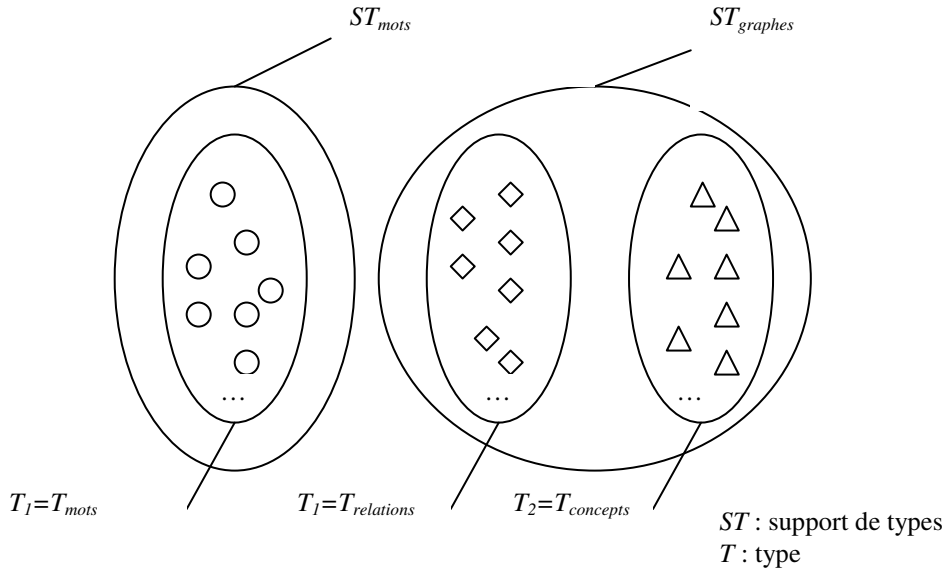


Figure 32 Deux supports de types

1.2 Support de vocabulaires d'un système de recherche d'information

Nous définissons les supports de vocabulaires qui permettent de modéliser des représentations. Ces supports possèdent l'avantage de définir les éléments qui constituent une représentation, leur expressivité et leur complexité et par conséquent l'expressivité de la représentation.

1.2.1 Vocabulaire d'un système de recherche d'information

Un vocabulaire *V* se compose à partir d'un ou plusieurs types T_i de *ST*. Ce même vocabulaire peut posséder une ou plusieurs pondérations restituant l'importance des éléments. Nous distinguons de fait deux axes pour classer les différents vocabulaires :

- Un vocabulaire *V* représente un vocabulaire simple ou complexe ;
  - *V* représente un **vocabulaire simple** si et seulement si il n'utilise qu'un seul T_i.
  - *V* représente un **vocabulaire complexe** si et seulement si il utilise plusieurs T_i.
- Un vocabulaire *V* désigne un **vocabulaire pondéré** si et seulement si V associe une ou plusieurs pondérations à un vocabulaire simple ou complexe.

1.2.1.1 Vocabulaire simple

a) Définition

Un vocabulaire simple *V* correspond à un seul type.

$$V \subseteq T_i \text{ avec } 1 \leq i \leq nst$$

On appelle unité de vocabulaires *uv* un élément de *V* tel que :

$$uv \in V \text{ , on a alors } uv \in T_i$$

b) Exemple

Soit un système de recherche d'information basé sur le support de types :

$$ST_{graphes}=(T_1, T_2) \text{ où } nst=2,\ T_1 = T_{concepts} \text{ et } T_2 = T_{relations}$$

Sur ce support, nous définissons le vocabulaire simple des concepts $V_{simpleConcepts}$ constitué d'unités de vocabulaires simples : les concepts. Ces concepts sont définis par le type $T_{concept}$ (cf. figure 33). Ce vocabulaire s'écrit alors :

$$V_{simpleConcepts} \subseteq T_{concepts} = T_1$$

Une unité de vocabulaire *uv* appartenant à $V_{simpleConcepts}$ constitue un des concepts de $T_{concepts}$, par exemple :

uv= C0817096(poumon)

1.2.1.2 Vocabulaire complexe

a) Définition

Un vocabulaire complexe *V* se compose de plusieurs types T_i d'un support de types *ST*. Un vocabulaire *V* formé de *nt* types s'écrit :

$$V \subseteq T_{fv(1)} \times T_{fv(2)} \ldots \times T_{fv(nt)} \text{ avec } nt > 1$$

et $fv : [1..nt] \longrightarrow [1..nst]$ la fonction qui détermine le type *fv(i)* du support *ST* utilisé par le i^{ème} type du vocabulaire complexe.

Une unité de vocabulaire *uv* se définit alors par :

$$uv \in V \text{ avec } uv = (v_1, .., v_{nt}) \text{ et } v_j \in T_{fv(j)},\ j \in [1..nt]$$

b) Exemple

Soit un système de recherche d'information basé sur le support de types :

$$ST_{graphes}=(T_1, T_2) \text{ où } nst=2,\ T_1 = T_{concepts} \text{ et } T_2 = T_{relations}$$

Le vocabulaire complexe $V_{complexeRelations}$, illustré sur la figure 33, représente des relations sémantiques entre concepts, ce vocabulaire s'écrit :

$$V_{complexeRelations} \subseteq T_{fv(1)} \times T_{fv(2)} \times T_{fv(3)} = T_{concepts} \times T_{relations} \times T_{concepts} \text{ avec } nt=3$$

Nous définissons donc la relation *fv* suivante :

$$fv : \begin{cases} 1 \longrightarrow 1 \\ 2 \longrightarrow 2 \\ 3 \longrightarrow 1 \end{cases}$$

Une unité de vocabulaire *uv* appartenant à $V_{complexeRelations}$ s'écrit par exemple :

uv= (C0817096 (poumon), est partie de, C0222762(cage thoracique))

où l'unité *uv* décrit le fait qu'un *poumon est une partie* de la *cage thoracique*.

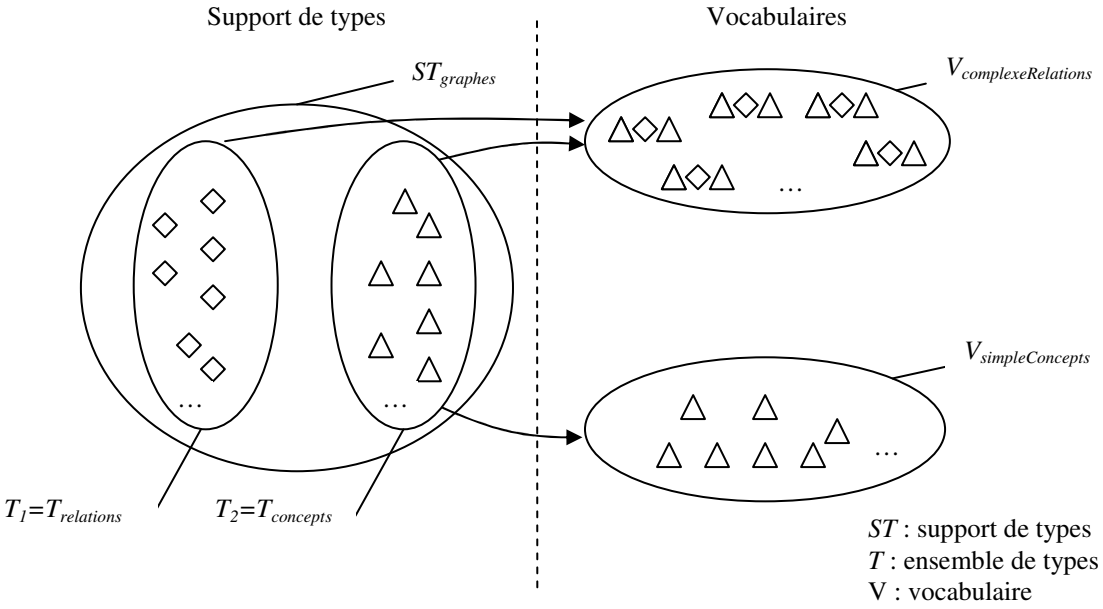


Figure 33 Deux vocabulaires : un simple $V_{simpleConcepts}$ et un complexe $V_{complexeRelations}$

1.2.1.3 Vocabulaire pondéré

a) Définition

Soit *V'* un vocabulaire simple ou complexe tel que défini en 1.2.1.1 et en 1.2.1.2. Un vocabulaire pondéré *V* consiste en l'association d'une ou plusieurs pondérations à ce vocabulaire *V'*.

Soit *np* le nombre de pondérations associées à *V,* nous définissons l'ensemble *V* par le produit cartésien entre un vocabulaire *V'* et un ensemble de *np* pondérations *P*.

$$V = V' \times P^{np} \text{ avec } P = \Re \text{ ensemble des réels et } np > 0$$

Une unité de vocabulaire *uv* se définit alors par :

$$V = \{uv \in V' \times P^{np}\} \text{ avec } uv = (uv', p) \text{ et } p \in P^{np}$$

b) Vocabulaire non pondéré correspondant

Nous définissons la fonction f_{np} qui pour un vocabulaire pondéré ou pour un sous-ensemble de ce vocabulaire retourne l'ensemble de vocabulaire simple ou complexe correspondant à ce vocabulaire.

$$f_{np}(V) = V'$$

Tel que $\forall uv' \in V'$ il existe au moins un *uv* tel que $\exists p \in P^{np} ; uv = (uv', p) \in V$

c) Exemple sur un vocabulaire simple

Soit $V'=V_{simpleConcepts}$, nous formons le vocabulaire pondéré $V_{poidsConcepts}$, à partir du vocabulaire simple *V'* et en utilisant une seule pondération (cf. tableau 4) :

$$V_{poidsConcepts} = V' \times P^1 \text{ avec } np=1$$

$$f_{np}\left(V_{poidsConcepts}\right)=V_{simpleConcepts}$$

Une unité de vocabulaire *uv* appartenant à $V_{poidsConcepts}$ se représente par exemple :

$$uv=(\ C0817096(poumon),\ 0.4)$$

avec $C0817096(p\ oumon)\in V'=V_{simpleConc\ epts}$

et $0.4\in P^1$ qui représente l'importance de ce concept dans un document.

d) Exemple sur un vocabulaire complexe

Soit $V'=V_{complexeRelations}$, nous formons le vocabulaire pondéré $V_{poidsRelations}$ à partir du vocabulaire complexe *V'* en utilisant deux pondérations (cf. tableau 4) :

$$V_{poidsRelations}=V'\times P^2 \text{ avec } np=2$$

$$f_{np}\left(V_{poidsRelations}\right)=V_{simpleRelations}$$

Une unité de vocabulaire *uv* appartenant à $V_{poidsRelation}$ se représente par exemple :

$$uv=(\ (C0817096(poumon),\ est\ partie\ de,\ C0222762(cage\ thoracique)),\ 0.4,\ 0.7)$$

avec (C0817096(poumon), est partie de, C0222762(cage thoracique))$\in V'=V_{complexeRelations}$

et $(0.4,0.7)\in P^2$ où l'une des pondérations représente l'importance de la relation dans un document et la deuxième reflète la confiance dans la détection de cette relation sur le document.

Vocabulaires	**np**	**Vocabulaires pondérés**
simple $V_{simpleConcepts}$ △ △ △ △ △ …	*1*	$V_{poidsConcepts}$ (△,0.2) (△,0.8) (△,0.2) (△,0.6) (△,0.4) …
complexe $V_{complexeRelations}$ △◇△ △◇△ △◇△ △◇△ …	*2*	$V_{poidsRelations}$ (△◇△,0.4,0.1) (△◇△,0.6,0.8) (△◇△,0.2,0.6)

Tableau 4 Exemples de vocabulaires pondérés

1.2.1.4 Résumé des vocabulaires

Un vocabulaire se compose de *nt* types et *np* pondérations :

$$V \subseteq T_{fv(1)} \times T_{fv(2)} \ldots \times T_{fv(nt)} \times P^{np}$$

avec $nt \geq 1$, $np \geq 0$ et $fv : [1..nt] \longrightarrow [1..nst]$

Un vocabulaire simple est formé d'un vocabulaire avec un seul type (*nt=1*) et aucune pondération (*np=0*). Un vocabulaire complexe est formé d'un vocabulaire avec plusieurs types (*nt>1*) sans pondération *(np=0*). Enfin un vocabulaire pondéré se compose d'un vocabulaire avec un ou plusieurs types (*nt*≥1) et avec une ou plusieurs pondérations (*np*≥1). Le tableau 5 résume ces trois sortes de vocabulaires.

Vocabulaire simple	*nt=1, np=0*	$V \subseteq T_i$
Vocabulaire complexe	*nt>1, np=0*	$V \subseteq T_{fv(1)} \times \ldots \times T_{fv(nt)}$
Vocabulaire pondéré	*nt≥1, np>0*	$V \subseteq V' \times P^{np}$ avec $P = \Re$

Tableau 5 Résumé des vocabulaires

1.2.2 Support de vocabulaires

a) Définition

On appelle support de vocabulaires *SV* une liste de vocabulaires utilisée par un système de recherche d'information. Ce support constitue un n-uplet formé de *nsv* vocabulaires que l'on ordonne de 1 à *nsv* :

$SV=(V_1, V_2, \ldots, V_{nsv})$ avec $nsv \geq 1$

b) Exemple

Un système de recherche d'information qui représente des graphes utilise un support de vocabulaires (cf. figure 34) constitué de deux ensembles, soit *nsv*=2, l'un représentant les concepts, l'autre représentant les relations entre ces concepts.

$$SV_{graphes} = (V_1, V_2) \text{ avec } nsv=2$$

où $V_1 = V_{poidsConcepts}$ et $V_2 = V_{poidsRelations}$

Nous écrivons directement : $SV_{graphes} = (V_{poidsConcepts}, V_{poidsRelations})$

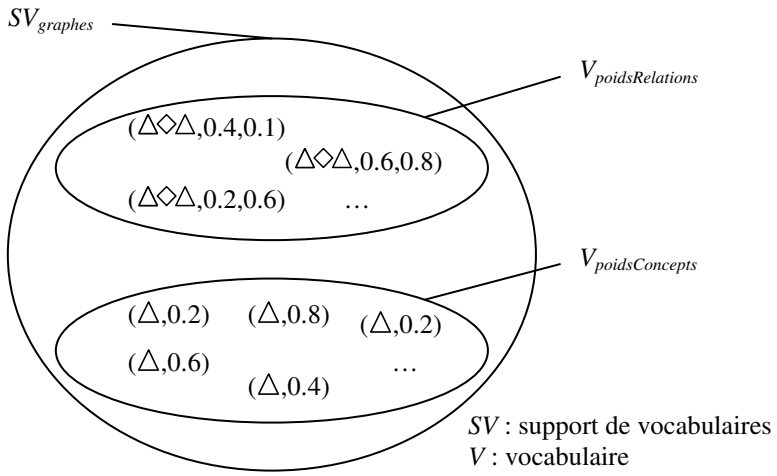


Figure 34 Un support de vocabulaires

1.3 Bilan

Un support de vocabulaires tel que nous le définissons dans la partie précédente caractérise l'expressivité d'une représentation basée sur ce support à deux niveaux :

- Au niveau de la complexité de la représentation, par l'utilisation de vocabulaires complexes et de différentes pondérations.
- Au niveau de la couverture des informations du document par la représentation de celui-ci, à travers différents vocabulaires qui donnent des points de vue ou des informations différentes sur son contenu.

La création d'un support de vocabulaires repose sur la définition au préalable d'un support de types *ST*. Ce support détermine le niveau sémantique de la représentation. Par exemple une représentation qui utilise des types concepts se situe à un niveau de représentation plus sémantique qu'une représentation qui n'utilise que des mots-clefs et qui se situe plus à un niveau lexical.

En recherche d'information, le modèle ne définit habituellement que les éléments du support de types *ST*. Comme la majorité des systèmes utilisent des représentations peu expressives, leur représentation se base souvent sur un seul vocabulaire simple avec une ou aucune pondération. De ce fait, la définition du modèle ne nécessite pas l'introduction de supports de vocabulaires, comme nous l'avons formellement défini. Au contraire, lors de l'utilisation de représentations expressives, les supports de vocabulaires deviennent très intéressants. Nous verrons par la suite comment un système de recherche d'information s'appuie sur la notion de support de vocabulaires.

2 Définition d'un système de recherche d'information

Un système de recherche d'information manipule un corpus de documents qu'il transpose à l'aide d'une fonction d'indexation en un corpus indexé. Ce corpus lui permet de résoudre des requêtes traduites à partir de besoins utilisateur. Un tel système repose sur la définition d'un modèle de recherche d'information *M* qui effectue ces deux transpositions et qui fait correspondre les documents aux requêtes.

La transposition d'un document en un document indexé repose sur un modèle de document. De même, la transformation du besoin utilisateur en requête repose sur un modèle de requête. Enfin, la correspondance entre une requête et des documents s'établit par une relation de pertinence. Le modèle *M* définit ces trois éléments.

2.1 Éléments manipulés par un système de recherche d'information

2.1.1 Corpus de documents

Nous disposons d'un corpus *C* de documents *d*. Nous notons *nc* sa cardinalité.

$$C = \{d\},\ \|C\| = nc\ .$$

2.1.2 Corpus Indexé

Le corpus indexé *CI* est constitué par l'ensemble des documents *di* indexés. La cardinalité de *CI* est *nc* :

$$CI = \{di\},\ \|CI\| = nc\ .$$

A chaque document *d* du corpus *C* correspond une indexation de ce document *di* qui représente le contenu du document *d* selon le modèle de documents. Le système de recherche d'information comprend une fonction d'indexation *ind* :

$$ind : C \longrightarrow CI\ .$$

Cette fonction permet de transposer le document *d* en son indexation *di*.

Dans un modèle vectoriel à base de mots-clefs muni d'une pondération de recherche d'information, cette relation correspond à :

$$ind(d) = di = \{(mot_1, poids_1), (mot_2, poids_2), ..., (mot_k, poids_k), ...\}$$

où $poids_k$ indique l'importance du mot_k dans le document *d*.

2.1.3 Requêtes

Le besoin d'information *b* représente la motivation de l'activité de recherche. Lorsque l'individu s'adresse directement au système de recherche, l'utilisateur formule le besoin d'information *b* sous forme d'une requête *q*. Cette requête représente de manière plus ou moins approximative le besoin d'information sous-jacent dans le modèle de requête. Nous supposons donc l'existence d'une fonction d'interprétation *inter* qui traduit un besoin d'information *b* en une requête *q* :

$$inter(b) = q$$

2.2 Modèle de recherche d'information

a) Définition

Un modèle *M* de recherche d'information se définit par un quadruplet formé d'un support de types *ST*, de deux supports de vocabulaires *SV* définis sur le support de types *ST* que l'on nomme *SVD* et

SVQ qui représentent le modèle de document et le modèle de requête, et d'une relation de correspondance *RC*.

$M = (ST, SVQ, SVD, RC)$ avec

$ST=(T_1, T_2, \ldots, T_{ns})$ $ns \geq 1$

$SVD=(V_1, V_2, \ldots, V_{nvd})$ avec $\forall i \in [1, nvd]$, V_i défini sur *ST*

$SVQ=(V_1, V_2, \ldots, V_{nvq})$ avec $\forall i \in [1, nvq]$, V_i défini sur *ST*

$RC = \{(q, di)\}$ *avec q* défini sur *SVQ* et *di* défini sur *SVD*

Nous remarquons que, dans de nombreux systèmes de recherche d'information *SVD* et *SVQ* se modélisent de la même façon.

b) Exemple

À l'aide des supports présentés précédemment, nous créons le modèle de recherche d'information $M_{graphes}$ basé sur les graphes :

$$M_{graphes} = (ST_{graphes}, SVD_{graphes}, SVQ_{graphes}, RC_{projection}) \text{ avec :}$$

- Le support de types :

$$ST_{graphes} = (T_{concepts}, T_{relations})$$

- Les modèles de document et de requête :

$$SVD_{graphes} = SVQ_{graphes} = SV_{graphes} = (V_{poidsConcepts}, V_{poidsRelations}), \text{ (cf. figure 34)}$$

- Et enfin le modèle de correspondance :

$$RC_{projection} = projection\ de\ graphes$$

2.3 Modèle de document

2.3.1 Modèle

a) Définition

Pour un modèle *M,* nous appelons support de vocabulaires de document *SVD* la liste des vocabulaires utilisés pour représenter les documents indexés. Ce support détermine le modèle de document. Le support *SVD* correspond à un support de vocabulaires constitué de *nvd* vocabulaires :

$$SVD = (V_1, V_2, \ldots, V_{nvd}) \text{ avec } nvd \geq 1$$

Nous distinguons deux types de modèles de document :

- *nvd=1 :* les modèles de document **mono-index** qui utilisent un seul ensemble de vocabulaires. Ces types d'index constituent ceux habituellement utilisés en recherche d'information textuelle. Un système d'indexation conceptuel n'utilise qu'un seul vocabulaire pour l'indexation, celui des concepts.

- *nvd>1 :* les modèles de document **multi-index** qui utilisent plusieurs ensembles de vocabulaires. Ces systèmes utilisent plusieurs vocabulaires pour représenter de façon multiple les documents : un système de recherche de vidéos peut utiliser d'une part un vocabulaire textuel pour décrire les scènes du film et d'autre part un vocabulaire visuel pour décrire les images de ce film.

b) Exemple mono-index

Dans un modèle de recherche d'information conceptuel $M_{conceptuel}$ où le modèle de document représente les documents à l'aide de concepts, le support d'indexation consiste en un mono-index $SVD_{concepts}$ tel que :

$$SVD_{concepts} = (V_{poidsConcepts}) \text{ avec } nvd = 1$$

c) Exemple multi-index 1

Dans un modèle de recherche d'information $M_{graphes}$ où le modèle de document représente les documents à l'aide de concepts et de relations entre concepts. Le support d'indexation consiste en un multi-index $SVD_{graphes}$ tel que :

$$SVD_{graphes} = SV_{graphes} = (V_{poidsConcepts}, V_{poidsRelations}) \text{ avec } nvd = 2$$

d) Exemple multi-index 2

Dans un modèle de recherche d'information $M_{multilingue}$ où le modèle de document représente les documents dans différentes langues (par exemple français, allemand et anglais) à l'aide d'un vocabulaire textuel français, d'un vocabulaire allemand et d'un vocabulaire anglais, le support d'indexation consiste en un multi-index $SVD_{multilingue}$ tel que :

$$SVD_{multilingue} = (V_{français}, V_{allemand}, V_{anglais})$$

avec $nvd = 3$ et $V_{francais} = T_{mots-clefs} \times P^1$, $V_{allemand} = T_{\text{schlüsselwort}} \times P^1$, $V_{anglais} = T_{key-words} \times P^1$,

où :

$T_{mots\text{-}clefs}$ représente un ensemble de mots-clefs en français,

$$T_{mots\text{-}clefs}=\{les, variétés, pomme, de, terre, \ldots\}$$

$T_{schlüsselwort}$ représente un ensemble de mots-clefs en allemand,

$$T_{schlüsselwort}=\{die, sorten, kartoffeln, \ldots\}$$

$T_{key\text{-}words}$ représente un ensemble de mots-clefs en anglais,

$$T_{key\text{-}words}=\{varieties, of, potatoes, \ldots\}$$

P^1 représente la pondération des mots-clefs pour la recherche d'information.

2.3.2 Représentation d'un document indexé

a) Définition

Un document indexé *di* se compose de *nvd* ensembles *DV* tels que :

$$di = (DV_1, \ldots, DV_i, \ldots, DV_{nvd}) \text{ avec } DV_i \subseteq V_i$$

DV_i forme un sous-ensemble de vocabulaires défini par la sélection dans V_i des éléments représentant le document. Un ensemble DV_i est constitué de nu_i unités de vocabulaires *uv*, tel que :

$$uv \in DV_i \text{ et } \|DV_i\| = nu_i$$

b) Exemple 1

Dans le support de vocabulaires de document $SVD_{graphes}$ la représentation *di* d'un document *d* se définit par :

$$di = (DV_{poidsConcepts}, DV_{poidsRelations}) \text{ avec}$$

$$DV_{poidsConcepts} \subseteq V_{poidsConcepts} \text{ et } DV_{poidsRelations} \subseteq V_{poidsRelations}$$

Une représentation de *di* pour un document s'écrit par exemple :

di=(

{(C0817096(poumon),0.4), (C0032225(plèvre),0.6), … },

{ ((C0817096(poumon), est partie de, C0222762 (cage thoracique)), 0.4,0.7),

((C0032225(plèvre), est partie de, C0222762(cage thoracique)), 0.4,0.7), …}

)

d	$di = (DV_{poidsConcepts}, DV_{poidsRelations})$	
Document	$DV_{poidsConcepts}$	$DV_{poidsRelations}$

La **plèvre** est une mince paroi double séparée par un liquide qui permet aux **poumons** de glisser doucement à l'intérieur de la **cage thoracique**.	{
	(C0817096(poumon),0.4)
	
	(C0032225(plèvre),0.6)
	
	(C0222762(cage thoracique),0.8)
	}
		{
		(C0817096(poumon), est partie de, C0222762 (cage thoracique), 0.4,0.7)
		
		(C0817096(poumon), est partie de, C0205076 (paroi thoracique), 0.3,0.1)
		
		(C0032225(plèvre), est partie de, C0222762(cage thoracique), 0.4,0.7)
		
		(*C0817096* (poumon), touche, C0032225(plèvre), 0.4,0.7)
		}

Tableau 6 Document indexé basé sur le modèle de document $SVD_{graphes}$

Nous donnons dans le tableau 6 la représentation d'un document *d*, à l'aide du modèle de document $SVD_{graphes}$. Dans ce document, d'une part le système détecte trois concepts car ces derniers correspondent à des termes de la phrase (en gras) ; d'autre part, le modèle utilise quatre relations dans l'index : trois de ces relations relient directement des concepts de la phrase et la quatrième relie un concept générique avec un concept de la phrase.

c) Exemple 2

Dans le support de vocabulaires de document $SVD_{multilingue}$, la représentation *di* d'un document *d* se définit par :

$$di = (DV_{français}, DV_{allemand}, DV_{anglais})$$

avec $DV_{français} \subseteq V_{français}$, $DV_{allemand} \subseteq V_{allemand}$ et $DV_{anglais} \subseteq V_{anglais}$.

Le tableau 7 donne la représentation d'un document *d* à l'aide du modèle de document $SVD_{multilingue}$.

d	$di = (DV_{français}, DV_{allemand}, DV_{anglais})$		
Document	$DV_{français}$	$DV_{allemand}$	$DV_{anglais}$

Radio d'un implant dentaire ou d'un plombage	{
	(Radio,0.25),
	(Implant,0.45),
	(Dentaire,0.25),
	(Plombage,0.85)
	}
		{
		(Röntgenbild, 0.13),
		(Zahnfüllung,, 0.16),
		(Zahnprothese, 0.48)
		}
			{
			(Xray, 0.25),
			(Dental, 0.54),
			(Implant, 0.25),
			(Filling, 0.95)
			}

Tableau 7 Document indexé basé sur $SVD_{multilingue}$

2.4 Modèle de requête

2.4.1 Modèle

Pour un modèle de recherche d'information *M*, on appelle support de vocabulaires de requête *SVQ* la liste des vocabulaires utilisés pour représenter les requêtes, ce support détermine le modèle des requêtes. Le support de vocabulaires de requêtes *SVQ* représente un support de vocabulaires constitué de *nvq* vocabulaires de *SV* :

$$SVQ = (V_1, ..., V_i, ..., V_{nvq}) \text{ avec } nvq \geq 1$$

2.4.2 Représentation

Une requête *q* se compose de *nvq* ensembles *QV* et s'écrit :

$$q = (QV_1, ..., QV_i, ..., QV_{nvq}) \text{avec} QV_i \subseteq V_i$$

QV_i forme un sous-ensemble de vocabulaire défini par la sélection dans V_i des éléments qui représentent la requête. Un ensemble QV_i est composé de nu_i unités de vocabulaires *uv* :

$$uv \in QV_i \text{ et } \|QV_i\| = nu_i$$

Ces représentations et ces modèles sont identiques, dans leur construction, à ceux proposés pour les documents dans la section précédente.

2.5 Modèle de correspondance

Le modèle de correspondance se base sur la définition d'une relation de correspondance entre les documents et les requêtes. La relation de correspondance *RC* définit une relation de pertinence entre un document et une requête donnée par :

$$RC = \{(q, di)\}$$

RC s'appuie sur la fonction de pertinence *Pert* qui, pour chaque document de la collection et chaque requête, calcule la pertinence du document vis-à-vis de la requête :

$$Pert : SVQ \times SVD \longrightarrow \Re$$
$$(q, di) \longrightarrow valeur$$

3 Récapitulatif des notations

nom	ensemble	cardinalité	élément	constitution	
Modèle de recherche d'information	***M***			*M=(ST, SVQ, SVD, FC)*	
Support de types	***ST***	*nst*	T_i	$ST=(T_1,\ldots, T_i, \ldots, T_{nst})$	
Type	*T*	|*T*|	*t*		
Vocabulaire	*V*	|*V*|	*uv*	$V \subseteq T_{fv(1)} \times \ldots \times T_{fv(nt)} \times P^{np}$	*nt* : nombre de type *np* : nombre de pondérations
Support de vocabulaires De document	***SVD***	*nvd*	V_i	$SVD = (V_1, \ldots, V_i, \ldots, V_{nvd})$	
Document indexé	*di*	*nvd*		$di = (DV_1, \ldots, DV_i, \ldots, DV_{nvd})$	
Sous-ensemble de vocabulaire	DV_i	nu_i	*uv*	$DV_i \subseteq V_i$	
Support de vocabulaires de requête	***SVQ***	*nvq*	V_i	$SVQ = (V_1, \ldots, V_i, \ldots, V_{nvq})$	
Requête	*Q*	*nvq*	*uv*	$Q = (QV_1, \ldots, QV_i, \ldots, QV_{nvq})$	
Sous-ensemble de vocabulaire	QV_i	nu_i	*uv*	$QV_i \subseteq V_i$	
Fonction de correspondance	***RC***			$RC = \{(q, di)\}$	

Tableau 8 Récapitulatif des ensembles

4 Bilan

Le cadre de recherche d'information proposé permet d'établir des modèles qui expriment plusieurs points de vue (modèle multi-index) plus ou moins complexes sur le document. Ce cadre met en avant la création de supports de vocabulaires qui permettent de décrire le modèle des documents et le modèle des requêtes. Le schéma de transposition des besoins en système de recherche d'information tel que décrit sur la figure 35 représente ces différents éléments. L'intérêt de ce cadre est de représenter des modèles divers dans un même formalisme. Cette représentation par l'utilisation de supports de vocabulaires met en avant l'expressivité des modèles et permet de positionner des modèles

les uns par rapport aux autres. L'utilisation des supports de vocabulaires permet de distinguer de nouveaux critères.

Dans le chapitre suivant nous proposons deux modèles d'expressivité similaires mais qui se distinguent sur de nouveaux critères liés à l'utilisation des supports de vocabulaires.

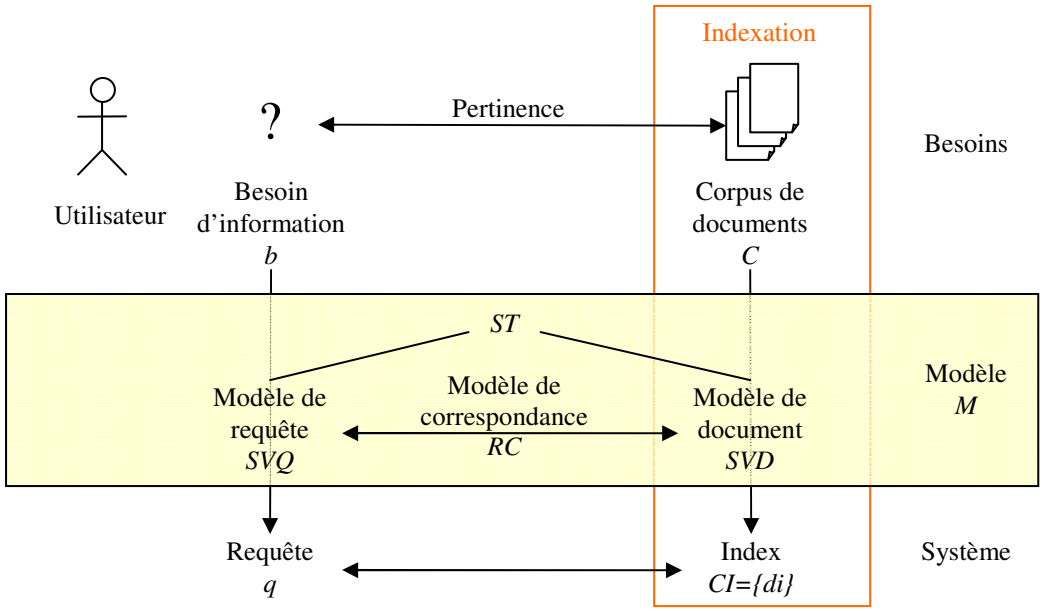


Figure 35 Traduction des besoins en un système de recherche d'information à l'aide du modèle proposé

Chapitre VII Deux Modèles Expressifs de Recherche d'Information

« Si l'homme n'a pas le pouvoir de modeler le monde à sa convenance, il a du moins celui de tailler des verres qui lui permettent de le faire apparaître à peu près comme il veut. » Georg Christoph Lichtenberg

Nous proposons d'utiliser des systèmes orientés précision. Pour modéliser l'expressivité de tels systèmes, nous employons des supports de vocabulaires. Nous définissons ici des modèles d'expressivité similaire mais qui abordent des approches différentes de l'utilisation des supports de vocabulaires, notamment à l'indexation.

Dans ce chapitre, nous reprenons le cadre présenté dans la partie précédente pour détailler deux modèles basés sur des structures complexes, l'une dérivée des graphes conceptuels, l'autre basée sur un modèle de graphe, inspiré des approches modèle de langue.

1 Positionnement des modèles

1.1 Choix des modèles

Améliorer la précision d'un système nécessite l'utilisation de représentations plus précises et plus complètes que les simples sacs de mots habituellement utilisés en recherche d'information.

Nous proposons de prime abord d'utiliser des types plus informatifs que les mots ou les termes. Pour cette raison nous utilisons des types de niveau sémantique. Nous utilisons des concepts plutôt que des mots ou des termes pour améliorer la précision tout en prévenant de la variation des termes et de la synonymie. De même, nous utilisons des relations sémantiques pour relier les concepts afin d'identifier les rôles que jouent les concepts dans les documents.

Nous proposons ensuite des modèles qui représentent le plus largement possible le contenu thématique d'un document. Pour cela, nous utilisons des supports basés sur plusieurs vocabulaires qui expriment plusieurs points de vue du document. Les modèles proposés ici se composent de vocabulaires qui expriment d'une part une vision conceptuelle et d'autre part une vision relationnelle du document. Ces deux points de vue sont essentiels pour obtenir des représentations plus complètes que les concepts seuls.

Enfin nous proposons l'utilisation de vocabulaires complexes plutôt que de vocabulaires simples. En effet de tels vocabulaires permettent de représenter des informations plus précises. Les modèles présentés ici s'appuient sur des structures de graphes qui utilisent des vocabulaires complexes, par exemple celui des relations qui se compose de concepts et d'étiquettes de relation.

Si les deux modèles que nous présentons utilisent des vocabulaires semblables, ils les exploitent de deux façons différentes.

1.2 Différenciation des deux modèles

Les deux modèles proposés se différencient par l'utilisation des supports de vocabulaires à l'indexation.

1.2.1 Portée des vocabulaires

Les deux modèles se basent sur une construction différente du support de vocabulaires (cf. figure 36). Si les vocabulaires de ces deux modèles utilisent des types identiques, la portée de ces vocabulaires diffère en fonction du modèle. Le modèle local est exhaustif, il utilise tout le vocabulaire définissable par les types composant les vocabulaires. Le modèle global est spécifique, il utilise seulement une partie du vocabulaire définissable par les types, la partie utilisée sur la collection.

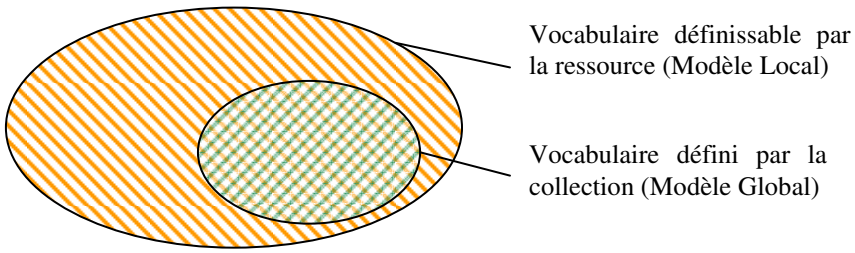


Figure 36 Détails des deux approches sur la portée des vocabulaires

1.2.2 Portée de la représentation de documents

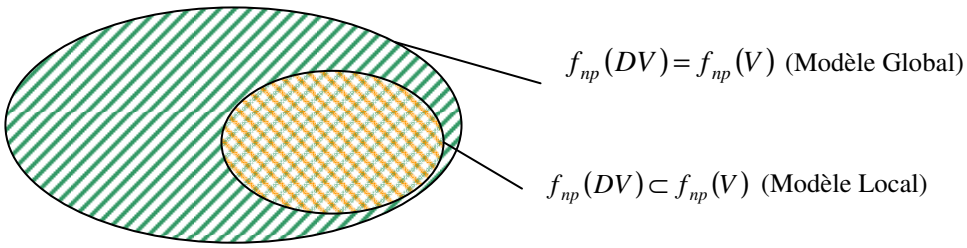


Figure 37 Détails des deux approches sur la portée des représentations par rapport aux vocabulaires

Sur ces supports, la portée de la représentation peut elle aussi être différente (cf. figure 37). Le modèle local forme une représentation **locale**. Au moment de l'indexation, elle ne représente le document qu'à l'aide d'une part limitée du support de vocabulaires (à l'exception des pondérations) :

$$f_{np}(DV) \subset f_{np}(V)$$

Le modèle global forme une représentation **globale** qui utilise la totalité des vocabulaires du support pour représenter le document. Elle utilise tous les éléments de vocabulaires (non pondérés) du support :

$$f_{np}(DV) = f_{np}(V)$$

Une représentation **locale** consiste en une représentation qui s'attache à refléter le contenu du document. Le modèle utilise seulement les éléments qui apparaissent dans le document ou qui peuvent se déduire à partir de celui-ci pour construire la représentation. La représentation du document

n'utilise donc qu'une partie du support de vocabulaires, celle correspondant au document. L'indexation se fait par interaction avec le support de vocabulaires (cf. figure 38).

Une représentation **globale** consiste en une représentation qui s'attache à refléter le document par rapport à son contexte. Elle voit le document comme une variation d'une représentation plus complète, ici la collection. La représentation du document forme une représentation complète sur les vocabulaires (sans les pondérations) du support de documents. L'indexation se fait à travers le support de vocabulaires (cf. figure 38).

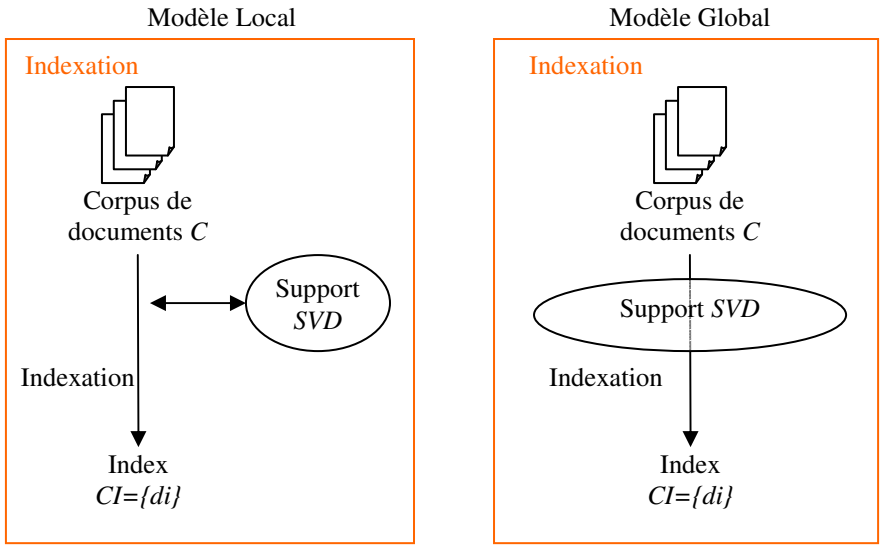


Figure 38 Comparaison des approches locale et globale

1.2.3 Correspondance

Les deux modèles se différencient aussi sur la fonction de correspondance. Le modèle local se base sur une comparaison des contenus, du document et de la requête. Le modèle global se base sur l'abstraction du contenu du graphe pour fournir un système plus génératif qui permet d'établir la capacité du document à générer la requête.

1.3 Synthèse

Nous utilisons donc deux modèles de représentation structurée des documents. Nous basons ces représentations sur des graphes constitués de relations binaires étiquetées et nous opposons deux méthodes de représentation, la figure 39 établie une synthèse de leur positionnement.

- Le Modèle Local consiste en une représentation stricte du contenu des documents basée sur un graphe de relations binaires. Il utilise des vocabulaires exhaustifs définissant toutes les unités de vocabulaires possibles.
- Le Modèle Global modélise le contenu des documents à travers le prisme du vocabulaire. Ce vocabulaire est spécifique car limité aux unités de vocabulaires de la collection.

L'espace formé par l'utilisation des supports de vocabulaires permet d'autres positionnements. Nous choisissons deux positionnements opposés dans cet espace mais qui correspondent à des positionnements usuels en recherche d'information (logique et modèle de langue).

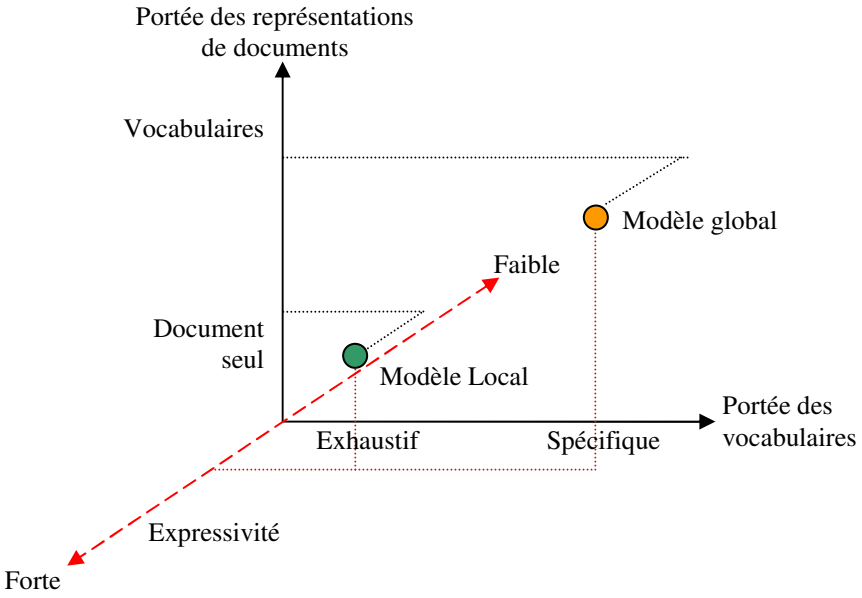


Figure 39 Positionnement des deux modèles par rapport à leur expressivité, et à l'utilisation du support de vocabulaires pour l'indexation

2 Éléments communs

2.1 Support de types

Les modèles proposés se basent sur des représentations dérivées de graphes. Nous utilisons un support *ST* commun aux deux modèles défini par 3 types $T_{concepts}, T_{relations}, T_{référents}$, que nous définissons par :

$$ST = (T_{concepts}, T_{relations}, T_{référents}) \text{ avec } nst = 3$$

- $T_{concepts}$ représente l'ensemble des **types de concepts** possibles,

  Exemple :

$T_{concepts}$={C0817096(poumon), C0032225(plèvre), C0018787(cœur), C0222762(cage thoracique), …}

- $T_{relations}$ représente l'ensemble des **noms de relations** binaires,

  Exemple :

$T_{relations}$={localisation, mesure, est partie de, touche…}

- $T_{référents}$ représente l'ensemble des **référents** de nœuds possibles,

  Exemple :

$T_{référents}$={#1, #2, #3…} ici le référent consiste en une référence au texte.

Nous utilisons des ressources telles que des thésaurus ou des ontologies pour la création de nos vocabulaires. Ces ressources permettent de définir les ensembles de types des deux modèles et fixent en partie l'expressivité des représentations.

$T_{concepts}$ est défini par l'ensemble des types de concepts d'une ressource tels un thésaurus ou une ontologie, par exemple le méta thésaurus UMLS.

$T_{relations}$ est défini par la même ressource ou par une autre, cette ressource fournit un ensemble de noms de nœuds de relations. Cette ressource permet de déterminer l'ensemble des nœuds concepts pouvant être reliés par un type de relations déterminé.

$T_{référents}$ définit l'ensemble des référents, sa définition peut consister en une liste d'autorité des référents et contient le référent générique '*'. Le référent permet de désigner l'élément correspondant au concept dans la représentation, par exemple un nom propre ou une référence.

Nous choisissons ici trois types de niveau sémantique, cela nous permet d'obtenir des supports de vocabulaires de niveau sémantique. Dans la suite de la thèse, l'instanciation des modèles déterminera plus précisément les ressources et les types retenus pour notre application.

2.2 Vocabulaire

Les deux modèles représentent des graphes. Par conséquent, ces deux modèles manipulent des vocabulaires proches. Nous définissons ici trois vocabulaires complexes utilisés par ces modèles :

- $V_{concepts}$, le **vocabulaire des concepts** formé de types de concepts et de référents :

$$V_{concepts} \subseteq T_{concepts} \times T_{référents} \text{ avec } nt = 2$$

Par exemple, une unité de vocabulaire *uv* appartenant à $V_{concepts}$ se note :

$$uv = (\ C0817096\ (poumon), \#2) \in V_{concepts}$$

avec $C0817096\ (poumon) \in T_{concepts}$ et $\#2 \in T_{référents}$ le référent du concept dans la représentation.

- $V_{couples}$, le **vocabulaire des couples** formé de couples de concepts :

$$V_{couples} \subseteq \left(V_{concepts}\right)^2 \subseteq \left(T_{concepts} \times T_{référents}\right)^2 \text{ avec } nt = 4$$

Par exemple, une unité de vocabulaire *uv* appartenant à $V_{couples}$ se note :

$$uv = (\ C0817096(poumon), \#2,\ C0222762(cage\ thoracique), \#3) \in V_{couples}$$

avec $(C0817096\ (poumon), \#2) \in V_{concepts}$ et $(C0222762(cage\ thoracique), \#3) \in V_{concepts}$

- $V_{relations}$, le **vocabulaire des relations** formé de couples de concepts associés à un nom de relation :

$$V_{relations} \subseteq V_{concepts}^{\ 2} \times T_{relations} \subseteq \left(T_{concepts} \times T_{référents}\right)^2 \times T_{relations} \text{ avec } nt = 5$$

Par exemple, une unité de vocabulaire *uv* appartenant à $V_{relations}$ se note :

uv=(C0817096 (poumon),#2, est partie de, C0222762(cage thoracique),#3) $\in V_{relations}$

avec *(C0817096 (poumon),#2)* $\in V_{concepts}$ *,(C0222762(cage thoracique),#3)* $\in V_{concepts}$

et *est partie de*$\in T_{relations}$.

3 Modèle Local ML

Dans cette partie nous modélisons les documents et les requêtes à l'aide de deux ensembles, l'un constitué des concepts, l'autre constitué des relations entre ces concepts. Ces deux ensembles représentent le contenu des documents sous forme de graphes. Ce modèle ne représente que les éléments contenus dans le document, il ne prend en compte l'information de contexte que par les pondérations sur ces éléments. Nous décrivons ce modèle à l'aide du cadre présenté dans le chapitre précédent. Nous montrons qu'une partie de notre modèle s'apparente à des graphes conceptuels. Nous détaillons ensuite la fonction de correspondance de notre modèle qui utilise la projection dans une première étape puis lui attribue un poids dans une seconde.

3.1 Modèle de recherche d'information

Nous définissons ici à l'aide de notre formalisme un modèle de recherche d'information basé sur des graphes composés de relations binaires étiquetées. Nous nommons *ML* ce modèle de recherche d'information :

$$ML = (ST, SVQ_{ML}, SVD_{ML}, RC_{ML})$$

Ce modèle utilise le support de types *ST* commun aux deux applications du modèle :

$$ST = (T_{concepts}, T_{relations}, T_{référents}) \text{ avec } nst = 3$$

À partir de ce support de types, nous établissons un modèle de document SVD_{ML} et un modèle de requête SVQ_{ML}. Ces deux modèles permettent de représenter les documents et les requêtes sous la forme de graphes. Nous établissons ensuite RC_{ML} la relation de correspondance entre ces graphes.

3.2 Modèle de document

Le modèle de document se définit à partir du support de vocabulaires de document SVD_{MOD}. Ce support se base sur deux vocabulaires, l'un représente les concepts pondérés, l'autre, les relations étiquetées pondérées :

$$SVD_{ML} = (V^{ML}_{conceptsDoc}, V^{ML}_{relationsDoc}) \text{ avec } nvd=2$$

Ces deux vocabulaires pondérés s'établissent à partir du vocabulaire complexe $V_{concepts}$, représentant les concepts, et du vocabulaire complexe $V_{relations}$ représentant les relations étiquetées :

- $V^{ML}_{conceptsDoc}$, le vocabulaire pondéré des concepts utilisés pour le modèle de document :

$$V^{ML}_{conceptDoc} = V_{concepts} \times P^{np_{conceptDoc}} \text{ avec } nt = 2 \text{ et } np = np_{conceptDoc}$$

Par exemple, dans le cas où $np_{conceptsDoc}=2$, un *uv* appartenant à $V^{ML}_{concepts}$ se note :

$$uv = (\text{ C0817096 (poumon)},\#2,\ 0.7,\ 0.9) \in V^{ML}_{concepts}$$

avec $uv' = (\text{ C0817096 (poumon)},\#2) \in V_{concepts}$ et $(0.7,\ 0.9) \in P^2$

- $V^{ML}_{relationsDoc}$, le vocabulaire pondéré des relations étiquetées utilisées pour le modèle de document :

$$V^{ML}_{relationsDoc} = V_{relations} \times P^{np_{relationDoc}} \text{ avec } nt = 5 \text{ et } np = np_{relationDoc}$$

Par exemple, dans le cas où $np_{relationDoc}=2$ un *uv* appartenant à $V^{ML}_{relations}$ se note :

$$uv = (\text{ C0817096 (poumon), \#2, est partie de, C0222762(cage thoracique), \#3, 0.3, 0.89}) \in V^{ML}_{relations}$$

avec $uv' = (\text{ C0817096 (poumon), \#2, est partie de, C0222762(cage thoracique), \#3}) \in V_{relations}$

et $(0.3, 0.89) \in P^2$

Nous ne fixons pas le nombre de pondérations utilisées par ce modèle ($np_{conceptDoc}$ et $np_{relationDoc}$), car il dépend de l'instanciation finale du modèle et du contexte dans lequel ce modèle est utilisé.

3.2.1 Représentation d'un document

a) Formalisme

La représentation *di* d'un document se forme de 2 ensembles :

$$di = \left(DV^{ML}_{concepts}, DV^{ML}_{relations}\right) \text{ où}$$

- $DV^{ML}_{concepts}$ est l'ensemble des **concepts pondérés** du document

$$DV^{ML}_{concepts} \subseteq V^{ML}_{conceptsDoc}, \text{ avec } uv_{concept} = \left(uv'_{concept}, p_{concept}\right) \in DV^{ML}_{concepts}$$

- $DV^{ML}_{relations}$ est l'ensemble des **relations étiquetées pondérées** du document

$$DV^{ML}_{relations} \subseteq V^{ML}_{relationsDoc}, \text{ avec } uv_{relation} = \left(uv'_{relation}, p_{relation}\right) \in DV^{ML}_{relations}$$

b) Contraintes

Dans ce modèle les sous-ensembles $DV^{ML}_{concepts}$ et $DV^{ML}_{relations}$ représentent les éléments qui apparaissent dans le document. Nous définissons trois contraintes sur ces ensembles :

- Unicité des concepts : un concept n'apparaît qu'une seule fois dans la représentation d'un document.

$$\text{Si } uv_{concept1} = \left(uv'_{concept}, p_{concept1}\right) \in DV^{MOD}_{concepts} \text{ et } uv_{concept2} = \left(uv'_{concept}, p_{concept2}\right) \in DV^{MOD}_{concepts}$$

$$\text{Alors } p_{concept1} = p_{concept2} \text{ et } uv_{concept1} = uv_{concept2}$$

- Unicité des relations : une relation n'apparaît qu'une seule fois dans la représentation d'un document.

$$\text{Si } uv_{relation1} = (uv'_{relation}, p_{relation1}) \in DV^{ML}_{relations} \text{ et } uv_{relation2} = (uv'_{relation}, p_{relation2}) \in DV^{ML}_{relations}$$

$$\text{Alors } p_{relation1} = p_{relation2} \text{ et } uv_{relation1} = uv_{relation2}$$

- Composition des relations : une relation étiquetée de la représentation d'un document est formée uniquement de concepts qui apparaissent dans cette représentation.

$$\text{Si } uv_{relation} \in DV^{ML}_{relations} \text{ avec } uv_{relation} = uv'_{concept1} \times uv'_{concept2} \times r \times p_{relation}$$

$$\text{alors } uv'_{concept1} \in f_{np}\left(DV^{ML}_{concepts}\right) \text{ et } uv'_{concept2} \in f_{np}\left(DV^{ML}_{concepts}\right)$$

Nous proposons dans le tableau 9 la représentation à l'aide de notre modèle d'un document dans lequel le système détecte trois concepts et trois relations entre ces concepts. Ce système ne détecte que des relations reliant des éléments de $DV^{ML}_{concepts}$ du fait de la contrainte posée sur le graphe.

d	$di = \left(DV^{ML}_{concepts}, DV^{ML}_{relations}\right)$	
Document	$DV^{ML}_{concepts}$	$DV^{ML}_{relations}$
La **plèvre** est une mince paroi double séparée par un liquide qui permet aux **poumons** de glisser doucement à l'intérieur de la **cage thoracique**.	{ (C0817096(poumon), #2, 0.4, 0.65) (C0032225(plèvre), #1, 0.6, 0.36) (C0222762(cage thoracique), #3, 0.8, 0.75) }	{ (C0817096(poumon), #2, est partie de, C0222762 (cage thoracique), #3, 0.4, 0.7) (C0032225(plèvre), #1, est partie de, C0222762(cage thoracique) ,#3), 0.32, 0.78) (C0817096 (poumon) ,#2, touche, C0032225(plèvre) ,#1, 0.82, 0.59) }

Tableau 9 Représentation d'un document par un modèle local

3.2.2 Modèle local versus modèle des graphes conceptuels

Le modèle local utilise les supports de vocabulaires pour proposer un système orienté précision. Il permet la représentation de modèles existants, notamment le modèle des graphes conceptuels. Nous montrons dans cette partie, la représentation syntaxique à l'aide du modèle local de graphes conceptuels.

Notre modèle peut utiliser un support de types identique au support des graphes conceptuels (limité à des relations binaires, et organisé par des treillis), il utilise les 3 ensembles $T_{concepts}$, $T_{relations}$, $T_{référents}$. Sur ce support, un graphe conceptuel $GC_d(C_d,R_d)$ représentant un document est un graphe bipartite composé de deux ensembles : l'ensemble des concepts C_d et l'ensemble des relations R_d.

- Un concept se représente par $[type : réf]$ où $type \in T_{concepts}$ décrit le type du concept, $réf \in T_{référents}$ décrit le référent.
- Une relation conceptuelle se représente par $(type(rel))$ où *rel* décrit un tuple de concepts flous de C et $type(rel)$ un type de relation définie dans $T_{relations}$. Le tuple *rel* s'ordonne et se

numérote de telle manière que chaque position corresponde à un rôle dont le sens dépend du type de relation $type(rel)$.

Nous montrons ici qu'un modèle *ML* sans pondération ($np_{conceptDoc}=0$ et $np_{relationDoc}=0$) forme un graphe conceptuel. Pour cela nous montrons que, sans pondération, les ensembles de concepts $DV_{concepts}$ et de relations $DV_{relations}$ qui constituent la représentation *di* d'un document correspondent aux ensembles de concepts C_d et de relations R_d qui constituent un graphe conceptuel $GC_d\ (C_d,R_d)$.

3.2.2.1 Concept

Nous montrons ici qu'un concept sans pondération tel que décrit dans notre modèle correspond à un concept du modèle des graphes conceptuels. Soit une unité de vocabulaire $uv_{concept}$ représentant un concept dans le modèle local avec ($np_{conceptDoc}=0$ et $np_{relationDoc}=0$), cette unité se note :

$$uv_{concept} \in DV_{conceptsDoc}$$
$$uv_{concept} \in T_{concepts} \times T_{référents}$$

L'unité de vocabulaire $uv_{concept}$ se compose de deux éléments, $type \in T_{concepts}$ le type du nœud et $réf \in T_{référents}$ le référent de ce nœud dans la phrase. Cette unité de vocabulaire correspond donc à un nœud concept tel que décrit dans les graphes.

Par exemple :

uv=(C0817096 (poumon),#35) s'écrit :

$uv = [C0817096\ (poumon):\#35]$ *dans le formalisme des graphes conceptuels.*

3.2.2.2 Relation

Nous montrons maintenant qu'une relation telle que décrite dans notre modèle correspond à un nœud relation des graphes conceptuels. Soit une unité de vocabulaire $uv_{relation}$ représentant une relation dans le modèle *ML* avec ($np_{conceptDoc}=0$ et $np_{relationDoc}=0$), cette unité s'écrit :

$$uv_{relation} \in DV_{relations}^{MOD}$$
$$uv_{relation} \in \left(T_{concepts} \times T_{référents}\right)^2 \times T_{relations}$$

L'unité de vocabulaire $uv_{relation}$ représentant une relation est constituée donc de deux nœuds $uv'_{concept1} \in f_{np}\left(DV_{concepts}^{ML}\right)$ et $uv'_{concept2} \in f_{np}\left(DV_{concepts}^{ML}\right)$ qui forment la relation binaire $rel = \left(uv'_{concept1}, uv'_{concept2}\right)$, et du type de cette relation $tr = type(rel) \in T_{relations}$. Cette unité de vocabulaire correspond donc à un nœud relation tel que décrit dans les graphes conceptuels.

Par exemple :

uv'=(C0817096 (poumon), #2, est partie de, C0222762(cage thoracique), #3)

s'écrit : $uv' = \left(type(rel)\right)$

Avec $rel = \left(C0817096\ (poumon), \#2, C0222762(cage\ thoracique), \#3\right)$

et $type\left(rel\right) =$ *est partie de* dans le formalisme des graphes conceptuels.

3.2.2.3 Bilan

Notre modèle local contient deux ensembles de nœuds, l'un constitué de concepts et l'autre de relations, qui forment un graphe bipartite, sans pondération. Ces deux ensembles s'expriment sous forme de graphes conceptuels qui utilisent seulement des relations binaires :

$$di = \left(DV^{MOD}_{concepts}, DV^{MOD}_{relations}\right) = GC_d\,(C_d, R_d)$$

avec $np_{conceptDoc}=0$ et $np_{relationDoc}=0$ et GC_d un graphe conceptuel

Nous présentons dans le tableau 10 une correspondance entre notre représentation des documents qui utilise deux ensembles de vocabulaires et la représentation du document à l'aide d'un graphe conceptuel. Nous remarquons ici que si le modèle local peut se représenter sous la forme de graphes conceptuels, du fait des contraintes il ne peut exprimer que des graphes sous forme normale. En effet, deux concepts identiques (référent et type) ne peuvent pas exister dans le modèle local alors que cela est possible dans le modèle des graphes conceptuels. Cette différence peut être comblée en ajoutant des témoins sur les nœuds du graphe dans le modèle local (Ounis et Pasca, 1998). Cependant, du fait de l'expression de la thématique et non du sens, avoir deux nœuds identiques dans un même graphe n'est pas aussi important en recherche d'information que dans d'autres domaines (linguistique). Par conséquent, notre modèle représente seulement des graphes conceptuels sous forme normale.

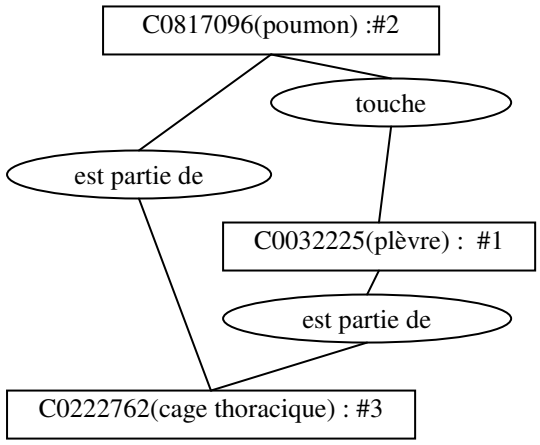
$di = \left(DV^{ML}_{concepts}, DV^{ML}_{relations}\right)$		$di = G_d\,(C_d, R_d)$
$DV^{ML}_{concepts}$	$DV^{ML}_{relations}$	
{ (C0817096(poumon), #2) (C0032225(plèvre), #1) (C0222762(cage thoracique), #3) }	{ (C0817096(poumon), #2, est partie de, C0222762 (cage thoracique), #3) (C0032225(plèvre), #1, est partie de, C0222762(cage thoracique) ,#3)) (*C0817096* (poumon) ,#2, touche, C0032225(plèvre) ,#1) }	

Tableau 10 Lien entre le modèle local et les graphes conceptuels pour la phrase du tableau 9

Nous pouvons généraliser la correspondance entre le modèle local et les graphes conceptuels. Nous considérons que les graphes tels que nous les proposons peuvent se décomposer en deux parties : une partie sans pondération qui constitue un graphe conceptuel et une partie qui ajoute des pondérations sur ces graphes. L'ensemble des opérations définies sur les graphes conceptuels est donc utilisable mais ne s'applique que sur le graphe sans prise en compte des pondérations. Les pondérations interviennent dans une étape ultérieure pour pondérer le résultat de l'opération considérée.

3.3 Modèle de requête

Dans le modèle local, le modèle de la requête se décrit de manière similaire au modèle des documents, ce modèle s'écrit alors :

$$SVQ_{ML} = \left(V^{ML}_{conceptsReq}, V^{ML}_{relationsReq}\right) \text{ avec } nvq=2$$

- $V^{ML}_{conceptsReq}$ dénote le vocabulaire pondéré des concepts utilisés pour le modèle de requête :

$$V^{ML}_{conceptsReq} = V_{concepts} \times P^{np_{conceptReq}} \text{ avec } nt=2 \text{ et } np=np_{conceptReq}$$

- $V^{ML}_{relationsReq}$ dénote le vocabulaire pondéré des relations utilisées pour le modèle de requête :

$$V^{ML}_{relationsReq} = V_{relations} \times P^{np_{relationReq}} \text{ avec } nt=5 \text{ et } np=np_{relationReq}$$

Une requête q se représente alors par :

$$q = \left(QV^{ML}_{concepts}, QV^{ML}_{relations}\right)$$

Sur cette représentation de la requête nous posons les mêmes contraintes que sur la représentation des documents, à savoir :

- Unicité des concepts : un concept n'apparaît qu'une seule fois dans une représentation de requête.
- Unicité des relations : une relation n'apparaît qu'une seule fois dans une représentation de requête.
- Composition des relations : une relation étiquetée de la représentation de la requête se forme uniquement de concepts qui apparaissent dans cette représentation.

3.4 Modèle de correspondance

En recherche d'information, comme présenté dans l'état de l'art, les modèles basés sur les graphes conceptuels utilisent la projection du fait de sa justification logique pour effectuer la correspondance entre les documents et la requête. De manière similaire, dans une première étape, nous utilisons la projection pour effectuer la correspondance entre la représentation des documents et celle des requêtes. Dans une deuxième étape, nous utilisons les pondérations pour qualifier cette projection, par le calcul de degrés de correspondance.

L'opération de projection permet de mettre en place un pré-ordre sur les graphes. La projection d'un graphe conceptuel H vers un graphe G correspond à une paire d'applications de H vers G, l'une sur les concepts et l'autre sur les relations des deux graphes, telles que :

- Elles conservent les arêtes et leur numérotation.
- Elles peuvent restreindre les étiquettes des sommets relation ou concept.

3.4.1 Pondération de la projection

En nous basant sur la projection, nous calculons un degré de correspondance δ entre concepts et entre relations comme dans (Mulhem *et al.*, 2001), mais nous instancions le degré de correspondance différemment.

Pour un graphe $q = \left(QV^{ML}_{concepts}, QV^{ML}_{relations}\right)$ représentant la requête et pour un graphe $di = \left(DV^{ML}_{concepts}, DV^{ML}_{relations}\right)$ représentant le document, nous définissons δ le degré de correspondance entre un concept $uv^{q}_{concept} = \left(type^{q}, réf^{q}, poids^{q}_{concept}\right)$ du graphe q et un concept $uv^{d}_{concept} = \left(type^{d}, réf^{d}, poids^{d}_{concept}\right)$ du graphe *di* par :

$$\delta\left(uv^{q}_{concept}, uv^{d}_{concept}\right) = \begin{cases} f_{concepts}\left(p^{q}_{concept}, p^{d}_{concept}\right) & \text{si } type^{d} \le type^{q} \text{ dans } T_{concepts} \\ & \text{et} \left(réf^{q} = * \text{ ou } réf^{q} = réf^{d}\right) \\ 0 & \text{sinon} \end{cases}$$

où $f_{concepts} : P^{np_{conceptReq}}, P^{np_{conceptDoc}} \longrightarrow [0...1]$ évalue une fonction des poids des deux concepts, qui rend compte de l'importance de la correspondance des deux concepts pour la tâche.

Le degré de correspondance entre une relation $uv^{q}_{relation}\left(uv^{q}_{concept1}, uv^{q}_{concept2}, tr^{q}, p^{q}_{relation}\right)$ du graphe de la requête q et une relation $uv^{d}_{relation}\left(uv^{d}_{concept1}, uv^{d}_{concept2}, tr^{d}\, p^{d}_{relation}\right)$ du graphe de document *di* se définit par :

$$\delta\left(uv^{q}_{relation}, uv^{d}_{relation}\right) = \begin{cases} f_{relations}\left(p^{q}_{relation}, p^{d}_{relation}\right) & \text{si } tr^{q} \le tr^{d} \text{ dans } T_{relations} \\ 0 & \text{sinon} \end{cases}$$

où $f_{relations} : P^{np_{relationReq}}, P^{np_{relationDoc}} \longrightarrow [0...1]$ évalue une fonction des poids de chaque relation, qui rend compte de l'importance de la correspondance des deux relations pour la tâche.

Les deux degrés de correspondance donnent des scores non nuls seulement aux éléments pouvant être reliés par une projection.

Pour une projection π, nous calculons le degré de correspondance de la projection, qui rend compte de la pertinence d'une projection du point de vue recherche d'information par :

$$\delta(\pi(q, di)) = \sum_{uv_{concept} \in QV^{MOD}_{concepts}} \delta\left(uv_{concept}, \pi\left(uv_{concept}\right)\right) + \sum_{uv_{relation} \in QV^{MOD}_{relations}} \delta\left(uv_{relation}, \pi\left(uv_{relation}\right)\right)$$

3.4.2 Relation de correspondance

Nous utilisons la projection pour établir la fonction de correspondance et la fonction de pertinence de notre modèle. Cependant, la majorité du temps la projection complète n'existe pas, comme proposé dans (Mulhem *et al.*, 2001), nous détectons indifféremment la projection d'un graphe ou d'un sous-graphe de la requête sur le graphe d'un document. La représentation d'un document *di* est pertinente pour une requête *q* s'il existe un sous-graphe de *q* qui se projette sur *di* :

$$RC_{ML} = \left\{q, di \middle| \exists\, \pi(\mathrm{q}', \mathrm{di}) \text{ avec } q' \text{ sous-graphe de } q\right\}$$

Puisqu'il peut exister plusieurs projections possibles, nous sélectionnons la projection qui obtient le plus fort degré de correspondance, l'utilisation de la somme dans le calcul du degré de correspondance favorisant les projections les plus complètes. En effet, cette projection sera considérée comme la plus intéressante pour la recherche d'information. La fonction de pertinence s'établit donc en calculant le degré de correspondance maximum entre GC_d et GC_q, :

$$Pert(q, di) = max_{\pi(q,di)}(\delta(\pi(q, di)))$$

3.5 Récapitulatif

nom	ensemble	cardinalité	constitution
Modèle de recherche d'information	$\boldsymbol{M_{ML}}$		$M_{ML} = (ST, SVQ_{ML}, SVD_{ML}, RC_{ML})$
Support de types	$\boldsymbol{ST}$	nst	$ST = (T_{concepts}, T_{relations}, T_{référents})$
Support de vocabulaires de document	$\boldsymbol{SVD_{ML}}$	$nvd=2$	$SVD_{ML} = (V^{ML}_{conceptsDoc}, V^{ML}_{relationsDoc})$
	$V^{ML}_{conceptDoc}$	$nt=2$ $np=np_{conceptDoc}$	$V^{ML}_{conceptDoc} = V_{concepts} \times P^{np_{conceptDoc}}$
	$V^{ML}_{relationsDoc}$	$nt=5$ $np=np_{relationDoc}$	$V^{ML}_{relationsDoc} = V_{relations} \times P^{np_{relationDoc}}$
Support de vocabulaires de requête	$\boldsymbol{SVQ_{ML}}$	$nvq=2$	$SVQ_{ML} = (V^{ML}_{conceptsReq}, V^{ML}_{relationsReq})$
		$nt=2$ $np=np_{conceptReq}$	$V^{ML}_{conceptsReq} = V_{concepts} \times P^{np_{conceptReq}}$
		$nt=5$ $np=np_{relationReq}$	$V^{ML}_{relationsReq} = V_{relations} \times P^{np_{relationReq}}$
Fonction de correspondance	$\boldsymbol{RC_{ML}}$		$RC = \{q, di \mid \delta(q, di) > 0\}$

Tableau 11 Récapitulatif du modèle local

3.6 Commentaires

Nous proposons ici un modèle local qui utilise une représentation proche des graphes conceptuels pour représenter les documents et les requêtes. La correspondance entre un document et une requête s'effectue par la projection qui compare le contenu des deux graphes. Nous nous servons ici de poids pour raffiner l'opération de projection en lui donnant une pondération. Ce modèle utilise un support de vocabulaires exhaustif, qui utilise tous les concepts et toutes les relations pouvant être définis. Il représente, sur ce support, le document uniquement avec les concepts et les relations représentatifs du contenu.

Le modèle se limite à l'utilisation de relations binaires entre concepts, cependant l'utilisation d'un modèle proche des graphes conceptuels permet de passer à des relations n-aires, notamment par l'utilisation d'un vocabulaire par arité de relation. Un autre intérêt de ce modèle vient de la facilité à intégrer différentes pondérations au niveau de la requête ou du document. L'introduction de ces pondérations nécessite seulement de déterminer les fonctions qui permettent d'établir les degrés de correspondance entre deux éléments. La proximité de ce modèle avec les graphes conceptuels permet de réutiliser des processus et des méthodes proposées pour les graphes conceptuels. En effet, les graphes conceptuels constituent un mécanisme d'expression riche et fortement étudié.

Ce modèle donne cependant un cadre de représentation strict, il peut difficilement intégrer des informations autres que celles représentées par les concepts et les relations. Ce modèle peut s'avérer

difficile à étendre notamment pour l'utilisation d'informations provenant d'autres sources que le document.

4 Modèle Global MG

Dans cette partie nous modélisons les documents à l'aide de trois vocabulaires. Le premier est constitué de concepts, le deuxième de relations binaires, et le dernier de relations étiquetées. Ce modèle représente un document sur l'ensemble des vocabulaires sachant que la portée de ces vocabulaires est limitée aux unités constituant la collection. Le modèle global donne une représentation statistique du document sur l'ensemble des vocabulaires que nous nommons *modèle de graphe*.

Nous détaillons le modèle global à l'aide du cadre présenté dans le chapitre précédent. Nous décrivons ici une extension des modèles de langue aux graphes qui s'intègre au sein de ce modèle. Enfin nous proposons une relation de correspondance entre les requêtes et les documents.

4.1 Modèle de graphe

Sur le texte, les modèles de langue capturent les régularités statistiques des mots au sein des phrases dans le but de former une représentation statistique de la langue. Ces modèles établissent la probabilité de distribution des chaînes de mots dans cette langue. En recherche d'information, les modèles de langue établissent la probabilité des chaînes de mots pour le sous-langage défini par un document. Ces modèles comparent le modèle de document M_D et la requête utilisateur par le calcul de la vraisemblance de la requête.

$$P(Q \mid M_D) \text{ avec } Q=mot1,mot2, \ldots ,motN.$$

Cette probabilité est calculée par décomposition de la chaîne de mots constituant la requête.

Les documents sont représentés ici par un ensemble de graphes, le modèle de langue doit alors s'adapter à ce format. Nous proposons d'établir un modèle de graphe M_D^g qui capture les régularités statistiques des graphes dans le but de former une représentation statistique du document. Pour comparer la requête et le document, le modèle de graphe évalue la probabilité du graphe Q de la requête par rapport au modèle de graphe M_D^g défini sur un document.

$$P(Q \mid M_D^g) \text{ avec } Q \text{ un graphe de requête}$$

Cette probabilité se calcule par décomposition du graphe en composants plus simples. Une probabilité est calculée pour les différents composants du graphe, c'est-à-dire les éléments des trois vocabulaires qui forment un graphe : les concepts, les couples et enfin les relations étiquetées. Ces probabilités prennent en compte le contexte du document, c'est-à-dire la collection, à l'aide de lissages.

4.2 Modèle de recherche d'information

Nous définissons ici un modèle basé sur la modélisation statistique de graphes. Nous nommons *MG* le modèle global utilisé par cette modélisation :

$$MG = (ST, SVQ_{MG}, SVD_{MG}, RC_{MG})$$

Ce modèle utilise le support de types *ST* commun aux deux modèles :

$$ST = (T_{concepts}, T_{relations}, T_{référents}) \text{ avec } nst = 3$$

A partir de ce support de types, nous établissons un modèle de document SVD_{MG} qui établit un modèle de graphe et un modèle de requête SVQ_{MG} qui décrit les éléments contenus dans le graphe de la requête. Enfin nous établissons la relation de correspondance RC_{MG} qui établit le lien entre les deux modèles.

4.3 Modèle de document

Le modèle de document est défini par le support de vocabulaires du document SVD_{MG}. Ce support se base sur trois vocabulaires, l'un représentant les concepts probables, le deuxième les couples probables et le dernier les relations étiquetées probables :

$$SVD_{MG} = (V^{MG}_{conceptsDoc}, V^{MG}_{couplesDoc}, V^{MG}_{relationsDoc}) \text{ avec } nvd=3$$

Ces trois vocabulaires constituent des vocabulaires pondérés. Ils s'établissent à partir des vocabulaires complexes communs aux deux modèles ; $V_{concepts}$ représentant les concepts, $V_{couples}$ représentant les couples de concepts et $V_{relations}$ représentant les relations, on a alors :

- $V^{MG}_{conceptsDoc}$, le vocabulaire pondéré des concepts utilisés pour le modèle de document :

$$V^{MG}_{conceptDoc} = V_{concepts} \times P^{np_{conceptDoc}} \text{ avec } nt = 2 \text{ et } np = np_{conceptDoc}$$

Par exemple, dans le cas où $np_{conceptDoc}=1$ un *uv* appartenant à $V^{MG}_{conceptsDoc}$ se note :

$$uv=(\text{ C0817096 (poumon)},\#2,\ 0.7) \in V^{MG}_{conceptsDoc}$$

avec uv'=(C0817096 (poumon),#2) $\in V_{concepts}$ et $(0.7) \in P^1$

- $V^{MG}_{couplesDoc}$, le vocabulaire pondéré des couples utilisés pour le modèle de document :

$$V^{MG}_{couplesDoc} = V_{couples} \times P^{np_{coupleDoc}} \text{ avec } nt = 4 \text{ et } np = np_{coupleDoc}$$

Par exemple, dans le cas où $np_{coupleDoc}=1$ un *uv* appartenant à $V^{MG}_{couplesDoc}$ se note :

$$uv=(\text{ C0817096 (poumon), \#2, C0222762(cage thoracique), \#3, } 0.3) \in V^{MG}_{couplesDoc}$$

avec $uv=(\text{ C0817096 (poumon), \#2, C0222762(cage thoracique), \#3)} \in V_{couples}$ et $(0.3) \in P^1$

- $V^{MG}_{relationsDoc}$, le vocabulaire pondéré des relations étiquetées utilisées pour le modèle de document :

$$V^{MG}_{relationsDoc} = V_{relations} \times P^{np_{relationDoc}} \text{ avec } nt = 5 \text{ et } np = np_{relationDoc}$$

Par exemple, dans le cas où $np_{relationDoc}=1$ un *uv* appartenant à $V^{MG}_{relationsDoc}$ se note :

uv=(C0817096 (poumon), #2, est partie de, C0222762(cage thoracique), #3, 0.89) $\in V^{MG}_{relationsDoc}$

avec *uv'=(C0817096 (poumon), #2, est partie de, C0222762(cage thoracique), #3)* $\in V_{relations}$

et *(0.89)* $\in P^{np_{relationDoc}}$

4.3.1 Représentation d'un document

a) Formalisme

La représentation d'un document *di* se forme donc de 3 ensembles :

$$di = \left(DV^{MG}_{concepts}, DV^{MG}_{couples}, DV^{MG}_{relations}\right) \text{ avec :}$$

- $DV^{MG}_{concepts}$ l'ensemble des **concepts probables** du document

$$DV^{MG}_{concepts} \subseteq V^{MG}_{conceptsDoc}, \text{ avec } uv_{concept} = \left(uv'_{concept}, p_{concept}\right) \in DV^{MG}_{concepts}$$

- $DV^{MG}_{couples}$ l'ensemble des **couples probables** du document

$$DV^{MG}_{couples} \subseteq V^{MG}_{couplesDoc}, \text{ avec } uv_{couple} = \left(uv'_{couple}, p_{couple}\right) \in DV^{MG}_{couples}$$

- $DV^{MG}_{relations}$ l'ensemble des **relations étiquetées probables** du document

$$DV^{MG}_{relations} \subseteq V^{MG}_{relationsDoc}, \text{ avec } uv_{relation} = \left(uv'_{relation}, p_{relation}\right) \in DV^{MG}_{relations}$$

b) Contraintes

Dans ce modèle de document, les sous-ensembles $DV^{MG}_{concepts}$ et $DV^{MG}_{relations}$ représentent les éléments qui apparaissent dans le document. Sur ces éléments nous posons les contraintes suivantes :

- Unicité des concepts : Un concept n'apparaît qu'une seule fois dans la représentation d'un document.

Si $uv_{concept1} = \left(uv'_{concept}, p_{concept1}\right)$ et $uv_{concept2} = \left(uv'_{concept}, p_{concept2}\right)$

Alors $p_{concept1} = p_{concept2}$ et $uv_{concept1} = uv_{concept2}$

- Unicité des couples : Un couple n'apparaît qu'une seule fois dans la représentation d'un document.

Si $uv_{couple1} = \left(uv'_{couple}, p_{couple1}\right)$ et $uv_{couple2} = \left(uv'_{couple}, p_{couple2}\right)$

Alors $p_{couple1} = p_{couple2}$ et $uv_{couple1} = uv_{couple2}$

- Unicité des relations : Une relation n'apparaît qu'une seule fois dans la représentation d'un document.

$$\text{Si } uv_{relation1} = (uv'_{relation}, p_{relation1}) \text{ et } uv_{relation2} = (uv'_{relation}, p_{relation2})$$

$$\text{Alors } p_{relation1} = p_{relation2} \text{ et } uv_{relation1} = uv_{relation2}$$

- Composition des couples : Un couple de la représentation d'un document est formé uniquement de concepts apparaissant dans cette représentation.

$$\text{Si } uv_{couple} \in DV^{MG}_{couples} \text{ avec } uv_{couple} = uv'_{concept1} \times uv'_{concept2} \times p_{relation}$$

$$\text{alors } uv'_{concept1} \in f_{np}\left(DV^{MG}_{concepts}\right) \text{ et } uv'_{concept2} \in f_{np}\left(DV^{MG}_{concepts}\right)$$

- Composition des relations : Une relation de la représentation d'un document est formée uniquement de couples apparaissant dans cette représentation.

$$\text{Si } uv_{relation} \in DV^{MG}_{relations} \text{ avec } uv_{relation} = uv'_{couple} \times tr \times p_{relation}$$

$$\text{alors } uv'_{couple} \in f_{np}\left(DV^{MG}_{couples}\right)$$

- Taille de la représentation : Le modèle global fournit une représentation du document sur l'ensemble des vocabulaires non pondérés :

$$\text{Pour les concepts : } f_{np}\left(DV^{MG}_{concepts}\right) = f_{np}\left(V^{MG}_{concepts}\right) = V_{concepts}$$

$$\text{Pour les couples : } f_{np}\left(DV^{MG}_{couples}\right) = f_{np}\left(V^{MG}_{couples}\right) = V_{couples}$$

$$\text{Pour les relations : } f_{np}\left(DV^{MG}_{relations}\right) = f_{np}\left(V^{MG}_{relations}\right) = V_{relations}$$

Nous proposons dans le tableau 12 la représentation d'un document à l'aide de ce modèle de document avec $np_{conceptDoc} = 1$, ce modèle attribue une valeur à tous les éléments du vocabulaire. Nous ne montrons ici qu'une partie de la représentation du document.

d	$di = \left(DV^{MG}_{concepts}, DV^{MG}_{couples}, DV^{MG}_{relations}\right)$		
Document	$DV^{MG}_{concepts}$	$DV^{MG}_{couples}$	$DV^{MG}_{relations}$
La **plèvre** est une mince paroi double séparée par un liquide qui permet aux **poumons** de glisser doucement à l'intérieur de la **cage thoracique**.	{ … (C0817096(poumon), #2, 0.4, 0.65) (C0032225(plèvre), #1, 0.6, 0.36) (C0222762(cage thoracique), #3, 0.8, 0.75) C0205076 (Paroi thoracique), C0015811(fémur) … }	{ … (C0817096(poumon), #2, C0222762 (cage thoracique), #3, 0.4, 0.6) (C0032225(plèvre), #1, C0222762(cage thoracique) ,#3), 0.32, 0.68) (*C0817096* (poumon) ,#2 , C0032225(plèvre) ,#1, 0.82, 0.18) (C0817096(poumon), C0205076 (paroi thoracique), 0.01,0.99) (C0817096(poumon), C0015811(fémur), 0.001,0.999) … }	{ … (C0817096(poumon), #2, est partie de, C0222762 (cage thoracique), #3, 0.4) (C0032225(plèvre), #1, est partie de, C0222762(cage thoracique) ,#3), 0.32) (*C0817096* (poumon) ,#2, touche, C0032225(plèvre) ,#1, 0.82) (C0817096(poumon), est partie de, C0205076 (Paroi thoracique), 0.01) … }

Tableau 12 Représentation d'un document par un modèle global

4.3.2 Modèle global versus modèle de graphe

Nous proposons ici un modèle de graphe qui s'intègre dans le modèle global.

Un graphe étiqueté simple se représente par deux ensembles *C* et *R* qui forment le graphe *G=(C,R)* où *C* représente l'ensemble des concepts formant le graphe, et *R* se définit par une fonction *type* qui associe à un couple de *C*×*C* l'ensemble des étiquettes *et* de ce couple (*type*(c_i, c_j) = *{et}* si c_i et c_j sont en relation par une relation étiquetée par les étiquettes de l'ensemble *{et}*, et ∅ sinon).

Nous considérons qu'un tel graphe est généré en deux étapes :

- Dans un premier temps les concepts *C* du graphe sont générés indépendamment les uns des autres, cette hypothèse correspond à celle habituellement utilisée dans les modèles de langue unigrammes.
- Ensuite sachant les concepts *C*, les relations *R* sont générées indépendamment les unes des autres.

Nous formulons ensuite les deux hypothèses suivantes sur le modèle de graphe :

- La génération d'un élément de *R* peut suivre plusieurs schémas, le premier considère les étiquettes comme complémentaires, le deuxième comme ambigües.
- La génération d'une étiquette pour une relation se décompose en deux étapes, la génération du couple de concepts, puis la génération de l'étiquette associée.

Un document indexé *di* contient les informations qui permettent de former un modèle de graphe du document. Ces informations prennent en compte les éléments qui apparaissent dans le document mais aussi ceux de son contexte.

Nous montrons ici que ce modèle de graphe s'intègre à la représentation d'un document. Cette représentation d'un document se définit par :

$$di = \left(DV^{MG}_{concepts}, DV^{MG}_{couples}, DV^{MG}_{relations}\right)$$

avec $DV^{MG}_{concepts}$ représentant l'ensemble des concepts probables $uv_{concept}$ du document *d*, $DV^{MG}_{couples}$ l'ensemble des relations probables entre les concepts $uv_{couples}$ et $DV^{MG}_{relations}$ l'ensemble des relations étiquetées probables $uv_{relations}$.

Du fait des caractéristiques de notre modèle de graphe, nous utilisons une seule probabilité par ensemble :

$$np_{conceptDoc} = 1,\ np_{coupleDoc} = 1,\ np_{relationDoc} = 1$$

Nous définissons M^{g}_{D} le modèle de graphe d'un document. Nous stockons ce modèle de graphe dans *di.* Nous obtenons donc le modèle suivant :

- Un **concept probable** se représente par :

$$uv_{concept} = \left(uv'_{concept}, P\left(uv'_{concept} \middle| M^{g}_{D}\right)\right)$$

avec $P\left(uv'_{concept} \middle| M^{g}_{D}\right) \in P^1$ et $uv'_{concept} \in V_{concepts}$

où $P\left(uv'_{concept} \middle| M^{g}_{D}\right)$ constitue la probabilité que le concept $uv'_{concept}$ soit généré par le modèle de graphe M^{g}_{D}.

- Un **couple probable** se représente par :

$$uv_{couple} = \left(uv'_{couple}, P_{couple}\right)$$

avec $uv'_{couple} = \left(uv'_{concept1}, uv'_{concept2}\right)$ et $P_{couple} = P\left(uv'_{couple} \middle| uv_{concept1}, uv_{concept2}, M^{g}_{D}\right)$

où P_{couple} dénote la probabilité que le couple de concepts formé de $uv_{concept1}$ et $uv_{concept2}$ soit relié dans le modèle de graphe.

- Une **relation probable** se représente par :

$$uv_{relation} = \left(uv'_{relation}, P_{rel}\right)$$

avec $uv'_{relation} = \left(tr, uv'_{couple}\right)$ et $P_{rel} = P\left(uv'_{relation} \middle| uv'_{couple}, M^{g}_{D}\right)$

où P_{rel} dénote la probabilité d'assigner l'étiquette *tr* au couple formé par $uv_{concept1}$ et $uv_{concept2}$ dans le modèle de graphe si le couple entre $uv_{concept1}$ et $uv_{concept2}$ existe.

Dans cette modélisation, la portée du vocabulaire se limite aux éléments qui appartiennent à la collection. La représentation d'un document constitue alors une représentation exhaustive sur toutes les unités des différents vocabulaires, et pour chacune de ces unités de vocabulaires la représentation contient les probabilités des éléments selon le modèle de graphe M^{g}_{D}. On peut donc écrire :

$$M^{g}_{D} = di$$

Le modèle de graphe s'intègre dans la représentation d'un document défini par le modèle global.

4.4 Modèle de requête

La requête est représentée par un graphe simple sans pondération, dont on cherche à établir la probabilité par rapport aux modèles de graphe des documents. Elle est modélisée par le support SVQ_{MG} tel que :

$$SVQ_{MG} = \left(V_{concepts}, V_{relations}\right) \text{ avec } nvq=2$$

Elle se représente par un ensemble de concepts et un ensemble de relations :

$$q = \left(QV^{MG}_{concepts}, QV^{MG}_{relations}\right)$$

La représentation de la requête respecte cependant la contrainte suivante :

- Composition des relations : une relation étiquetée de la requête est formée uniquement de concepts apparaissant dans la requête.

Le modèle de requête définit alors un graphe étiqueté $G_Q=(C_Q,R_Q)$ tel que G_Q soit équivalent à q, où C_q représente l'ensemble des concepts de la requête et R forme une fonction qui associe à un couple $C\times C$ l'ensemble des étiquettes *et* de ce couple. Nous montrons un tel graphe sur la figure 40.

$$q = G_Q \text{ avec } C_Q = QV^{MG}_{concepts} \text{ et } R_Q = QV^{MG}_{relations}$$

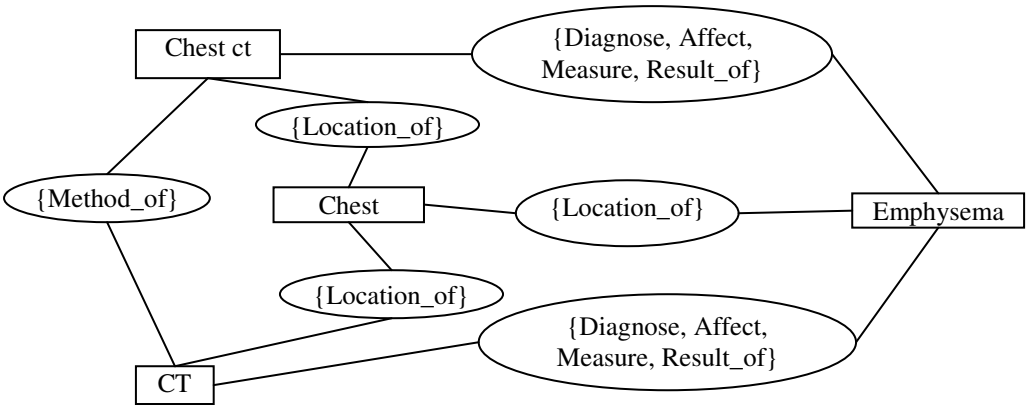


Figure 40 Graphe correspondant à la requête 'Show me chest CT images with emphysema'

4.5 Modèle de correspondance

Le modèle global propose de construire un modèle de graphe représentant chaque document. La requête quant à elle se représente par un graphe. Nous effectuons la correspondance en calculant la probabilité qu'un processus génératif basé sur le modèle de graphe du document génère le graphe de la requête (score de vraisemblance de la requête).

4.5.1 Génération de graphe

La représentation d'un document contient des informations sur les probabilités des unités de vocabulaires sachant le modèle de graphe du document ; à l'identique des modèles de langue sur le

texte qui calculent la probabilité de générer les phrases de la requête. Nous calculons ici la probabilité de générer le graphe de la requête :

$$P\left(G_Q \middle| M_D^g\right)$$

Le calcul de cette probabilité dépend du modèle de graphe choisi. Nous reprenons le processus génératif décrit en 4.3.2, nous considérons que chaque représentation de la requête se génère par un processus en deux étapes. La première étape produit l'ensemble des concepts. La seconde étape produit les relations étiquetées entre ces concepts. Nous décomposons donc la probabilité du graphe en :

$$P\left(G_Q \middle| M_D^g\right) = P\left(C_Q \middle| M_D^g\right) \times P\left(R_Q \middle| C_Q, M_D^g\right)$$

Nous posons ensuite l'hypothèse que les concepts sont générés indépendamment les uns des autres, comme l'effectuent en général les modèles de langue unigrammes. La probabilité $P\left(C_Q \middle| M_D^g\right)$ se calcule alors par :

$$P\left(C_Q \middle| M_D^g\right) = \prod_{c \in C_Q} P\left(c \middle| M_D^g\right)$$

La probabilité d'un concept se réécrit :

$$P\left(c \middle| M_D^g\right) = P\left(uv'_{concept} \middle| M_D^g\right) \text{ avec } uv'_{concept} \in V_{concept}$$

ce qui correspond à la probabilité stockée dans $DV_{concepts}^{MG}$.

De même, pour simplifier les calculs, nous posons l'hypothèse que, conditionnées par les concepts, les probabilités des relations deviennent elles aussi indépendantes.

$$P\left(R_Q \middle| C_Q, M_D^g\right) = \prod_{R_Q} P\left(type(r) = \{tr\} \middle| C_Q, M_D^g\right)$$

Nous posons ensuite l'hypothèse qu'une relation ne dépend que de la probabilité des concepts qu'elle relie. Sa probabilité s'écrit alors :

$$P\left(R_Q \middle| M_D^g\right) = \prod_{R_Q} P\left(type(r) = \{tr\} \middle| c1, c2, M_D^g\right) \text{ avec } r = (c1, c2)$$

Nous considérons ensuite qu'une relation apparaît dans la requête si le processus génère toutes ses étiquettes ; une relation forme une intersection d'étiquettes, elles doivent toutes apparaître pour que la relation existe. La probabilité précédente se décompose alors en :

$$P\left(type(r) = \{tr\} \middle| c1, c2, M_D^g\right) = \prod_{tr \in type(r)} P\left(type(r) = tr \middle| c1, c2, M_D^g\right)$$

La probabilité d'une relation étiquetée s'écrit alors :

$$P\left(type(r) = tr \middle| c1, c2, M_D^g\right) = P\left(uv'_{relation} \middle| uv'_{concept1}, uv'_{concept2}, M_D^g\right)$$

Enfin, nous choisissons soit d'estimer directement $P\left(type\left(r=tr\middle|c1,c2,M_D^g\right)\right)$, soit de décomposer cette probabilité en deux probabilités :

$$P\left(type(r)=tr\middle|c1,c2,M_D^g\right)=P\left(couple(c1,c2)=1\middle|c1,c2,M_D^g\right)\times P\left(type(r)=tr\middle|couple(c1,c2)=1,M_D^g\right)$$

La probabilité d'un couple s'écrit :

$$P\left(couple(c1,c2)\middle|c1,c2,M_D^g\right)=P\left(uv'_{couple}\middle|uv'_{concept1},uv'_{concept2},M_D^g\right)$$

et celle d'une relation étiquetée sachant le couple par :

$$P\left(type(r)=tr\middle|couple(c1,c2)=1,M_D^g\right)=P\left(uv'_{relation}\middle|uv'_{couple},M_D^g\right)$$

Ces deux dernières probabilités correspondent à celles exprimées dans le modèle de document.

4.5.2 Relation de correspondance

Nous utilisons la probabilité de vraisemblance de la requête pour établir la pertinence pour les documents. Cette probabilité décrit la capacité du modèle de graphe à générer le graphe de la requête.

$$pert(q,di)=P\left(G_q\middle|M_D^g\right)=P(q|di)$$

avec q graphe de requête et $M_D^g=di$ modèle de graphe.

Nous considérons un document *di* comme pertinent si la probabilité que son modèle de graphe M_D^g générant la requête donne un résultat supérieur à la probabilité de générer la requête sans information sur le document.

$$RC_{MG}=\left\{q,di\middle|P(q|di)>P(q|\phi)\right\}$$

4.6 Récapitulatif

nom	ensemble	cardinalité	constitution
Modèle de recherche d'information	$\boldsymbol{M_{MG}}$		$M_{MG} = (ST, SVQ_{MG}, SVD_{MG}, RC_{MG})$
Support de types	$\boldsymbol{ST}$	nst	$ST = (T_{concepts}, T_{relations}, T_{référents})$
Support de vocabulaires de document	$\boldsymbol{SVD_{MG}}$	$nvd=2$	$SVD_{MG} = (V^{MG}_{conceptsDoc}, V^{MG}_{coupleDoc}, V^{MG}_{relationsDoc})$
	$V^{MG}_{conceptDoc}$	$nt=2$ $np=np_{conceptDoc}$	$V^{MG}_{conceptDoc} = V_{concepts} \times P^{np_{conceptDoc}}$
	$V^{MG}_{coupleDoc}$	$nt=4$ $np=np_{coupleDoc}$	$V^{MG}_{coupleDoc} = V_{couples} \times P^{np_{coupleDoc}}$
	$V^{MG}_{relationsDoc}$	$nt=5$ $np=np_{relationDoc}$	$V^{MG}_{relationsDoc} = V_{relations} \times P^{np_{relationDoc}}$
Support de vocabulaires de requête	$\boldsymbol{SVQ_{MG}}$	$nvq=2$	$SVQ_{MG} = (V_{concepts}, V_{couples}, V_{relations})$
Fonction de correspondance	$\boldsymbol{RC_{MG}}$		$P(G_q \mid M^g_D)$

Tableau 13 Récapitulatif du modèle global

4.7 Discussion

Nous proposons un modèle global qui utilise un modèle de graphe pour représenter les documents. La correspondance entre un document et une requête s'effectue en évaluant la probabilité que le modèle de graphe génère la représentation de la requête. Ce modèle utilise un support de vocabulaires spécifique, limité aux concepts et aux relations de la collection. Il représente le document à l'aide de l'ensemble des unités de vocabulaires de ce support.

Le modèle global constitue un modèle plus flexible que le modèle local au niveau des éléments pris en compte. Il peut facilement utiliser des informations redondantes. Par exemple, les couples et les relations contiennent des informations redondantes. Par contre, en fonction du modèle de graphe choisi, la génération de ces deux éléments peut ne pas se voir attribuer de probabilité. Ici le processus génératif détermine quels éléments prendre en compte et comment les prendre en compte alors que le modèle local pondère le contenu du document. Si ce modèle rend compte plus facilement de certaines informations, il reste plus difficilement extensible à des relations n-aires, qui doivent trouver leur place dans le modèle de graphe qui devient plus complexe.

Le modèle global est plus complexe que le modèle local, car il donne une pondération à tous les éléments de chaque vocabulaire. Cependant, il représente mieux le document qu'il ne considère plus comme une entité indépendante des autres, mais comme une variation d'un modèle de graphe plus général basé sur la collection.

Le modèle global permet de dépasser la simple représentation du contenu car il établit un modèle de document définissant un ensemble de probabilités sur l'ensemble du vocabulaire du document. Le

calcul de ces probabilités peut donc prendre en compte des informations provenant de la collection ou d'autres sources d'information.

Si le modèle doit prendre en compte des événements annexes, par exemple des scores d'extraction sur les graphes, ces éléments nécessitent l'adaptation du modèle. Pour l'instant nous n'avons pas adapté le modèle de graphe pour qu'il prenne en compte explicitement les problèmes d'extraction.

5 Conclusion

Cette partie a présenté deux modèles de représentation d'expressivité proches, basés sur la notion de graphes. Ces représentations de documents sont toutes deux expressives et basées sur des vocabulaires proches représentant d'une part la vision conceptuelle du document et d'autre part la vision relationnelle.

Cependant, ces deux modèles s'opposent sur leurs définitions des vocabulaires : le modèle local utilise tout le vocabulaire pouvant se créer à partir des types utilisés par ce vocabulaire et le modèle global utilise le vocabulaire restreint à celui utilisé par la collection. Ces modèles diffèrent aussi par l'utilisation des vocabulaires lors de la représentation d'un document. Le modèle local n'utilise que le vocabulaire détecté dans le document pour représenter celui-ci, le modèle global utilise quant à lui la totalité du vocabulaire pour définir le contenu du document au sein de la collection. Enfin ces modèles utilisent deux modèles de correspondance différents. Le modèle local se base sur une méthode de correspondance qui compare le contenu de la requête avec celui des documents, alors que le modèle global se base sur la probabilité que le modèle de document génère la représentation de la requête.

Ces modèles forment deux conceptions différentes de la pertinence en recherche d'information. Le modèle local considère les documents pertinents s'ils possèdent une intersection de contenu avec la requête ; la pertinence s'améliore en fonction de l'augmentation de cette intersection. Le modèle global exploite de son côté les régularités statistiques du document pour déterminer si la requête se rapproche plus ou moins de ces régularités, c'est-à-dire si elle suit le même langage graphique. Ces deux modèles montrent que les supports de vocabulaires permettent de modéliser de façon uniforme des systèmes orientés précision très différents.

En recherche d'information textuelle, les modèles de type modèle de langue qui permettent la prise en compte des informations, autres que le contenu du document, montrent de meilleurs résultats que les approches basées sur des représentations du contenu telles que les méthodes vectorielles utilisant le *tf.idf*. Nous vérifions par la suite si ces résultats s'appliquent aussi aux graphes.

Bilan

Dans cette partie nous avons défini un cadre dans lequel s'exprime deux modèles possédant une expressivité forte. Ce cadre introduit la notion de support de vocabulaires. Ce support se base sur la définition au préalable d'un support de type qui détermine les types utilisés par le système. Un support de vocabulaires se compose de plusieurs vocabulaires créés à l'aide d'un ou de plusieurs des types définis dans le support de type associé à des pondérations.

Les supports de vocabulaires d'un modèle, celui utilisé pour modéliser les documents et celui utilisé pour modéliser la requête, permettent de définir l'expressivité du modèle de recherche d'information. L'état de l'art ayant fortement exploré l'axe de l'expressivité des représentations, nous choisissons d'employer des représentations expressives à base de graphes de niveau sémantique. Les modèles proposés possèdent donc des supports de vocabulaires proches et d'expressivité similaire, qui utilisent tous les deux des vocabulaires complexes et pondérés. Ils se fondent par ailleurs tous les deux sur le même support de type.

Ayant des expressions similaires, ces deux modèles explorent le plan formé par les axes décrivant la portée du support et la portée des représentations de documents. Sur ces axes ces deux modèles utilisent des méthodes antinomiques. Nous avons défini les vocabulaires du modèle local à partir de tous les éléments accessibles à partir des types. Dans le modèle global nous avons limité les vocabulaires aux éléments définissables et rencontrés sur la collection. Les représentations du modèle local n'utilisent que le vocabulaire du document alors que celles du modèle global utilisent les vocabulaires en entier.

Nos choix sur le support ont des implications dans le choix du type de modèle utilisé. Le modèle local se base sur un modèle de graphe utilisant la projection pour faire correspondre les documents. Le modèle global quant à lui fournit un modèle inspiré des modèles de langue qui calcule la probabilité de générer les représentations des requêtes. Ces deux modèles montrent donc que les supports de vocabulaires permettent de modéliser différents types de modèles et par conséquent de les comparer plus facilement. Cette facilité est d'autant plus importante sur l'expressivité car les supports de vocabulaires mettent en avant cette notion.

L'intérêt de ces deux modèles est qu'ils peuvent s'appliquer sur de nombreux média ou domaines, à condition qu'une ressource permette de définir les types concepts, relations et référents définis par le support de type. Cette application à différents domaines nécessite tout de même de fixer un certain nombre de paramètres. D'une part, au niveau des pondérations, l'utilisation du modèle local sur un domaine donné nécessite de déterminer le nombre de pondération, et l'utilisation du modèle global nécessite de détailler les estimations qui permettent les obtenir. D'autre part, au niveau de la correspondance, le modèle local nécessite de compléter les fonctions définissant le degré de correspondance entre concepts et entre relations. Enfin l'instanciation des deux modèles nécessite de déterminer la fonction d'indexation et la fonction d'interprétation de la requête de telle sorte qu'elles fournissent les représentations adéquates des documents et des requêtes.

Dans la suite nous proposons d'instancier ces deux modèles pour pouvoir les utiliser sur des textes dans un domaine précis.

PARTIE 4 : PROCESSUS D'INDEXATION POUR DES MODELES EXPRESSIFS

Introduction

La partie précédente présente deux modèles de recherche d'information qui se basent sur des représentations expressives. La définition de ces modèles reste générale. Ils peuvent s'appliquer sur différents domaines et sur différents médias à condition que ces derniers permettent l'utilisation de concepts et de relations sémantiques. Nous représentons cet espace comme sur le tableau 14. Nous nous intéressons pour notre part à l'application de ces modèles au média textuel sur le domaine médical et dans un contexte multilingue.

médias	textes				images			vidéos	
domaines	médicaux	manuels techniques	web	…	personnelles	médicales	…	journaux télévisées	…

Tableau 14 Exemple de support et de domaine possible pour l'application des modèles

Nous détaillons dans un premier temps les caractéristiques propres à l'application des deux modèles sur le texte. Nous avons décomposé l'application de ces modèles en deux étapes. Ces étapes font intervenir une représentation intermédiaire des documents constituée d'un ensemble de représentations de niveau sémantique, chacune correspondant à une phrase. Ces représentations suivent un modèle exprimé à partir d'un support de vocabulaires. Nous avons réalisé la fonction d'indexation et la fonction d'interprétation en deux phases séparées : la première consiste à détecter la représentation de chaque phrase, elle permet la création de la représentation intermédiaire. La seconde étape consiste en la création des représentations des documents et des requêtes à partir de cette représentation intermédiaire. Nous fournissons les caractéristiques du support de vocabulaires de la représentation intermédiaire, ce support intègre un score de confiance qui permet d'évaluer la qualité de ses constituants. Nous proposons ensuite les méthodes qui permettent de créer les deux modèles à partir de cette représentation intermédiaire du document.

Pour toute application des deux modèles sur du texte, la représentation intermédiaire et les méthodes de création des modèles à partir de cette représentation ne dépendent pas du domaine (cf. figure 41). L'application des deux modèles sur un domaine textuel particulier ne consiste alors plus qu'à fournir les méthodes qui permettent de créer la représentation intermédiaire des documents sur ce domaine ; domaine sur lequel nous devons définir concrètement le support de type utilisé. La qualité de la représentation intermédiaire est d'autant plus importante que tous les processus aboutissant à la création de nos deux modèles s'appuient sur cette représentation.

Nous détaillons donc dans un second temps les méthodes qui permettent de créer la représentation intermédiaire à partir de textes médicaux. Nous définissons précisément les types utilisés à l'aide de la source de connaissances UMLS. Nous détaillons ensuite plusieurs méthodes pour détecter les représentations intermédiaires sur les textes médicaux.

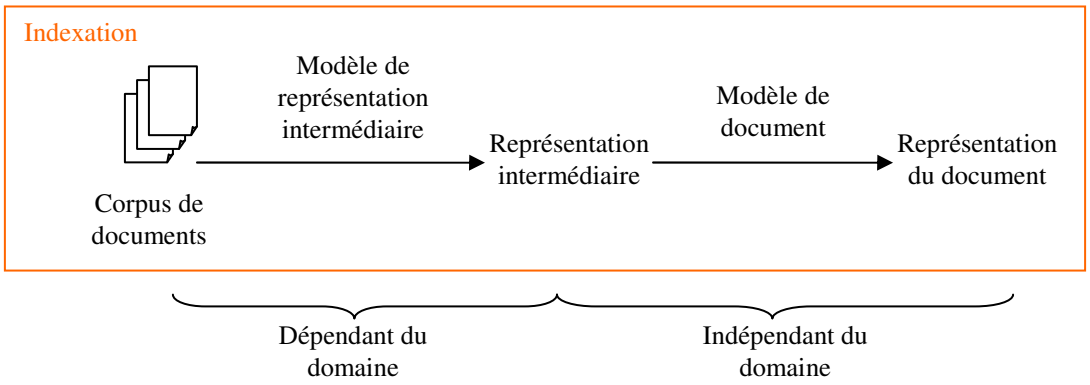


Figure 41 Décomposition de la fonction d'indexation

Chapitre VIII Application au Texte

« S'il n'y a pas de solution c'est qu'il n'y a pas de problème. » Jacques Rouxel (Extrait de la BD Les Shadoks)

Nous présentons dans cette partie les étapes du processus qui permettent de construire les deux modèles proposés à partir de textes. Ce processus découpe la création de nos modèles en deux étapes par l'utilisation d'une représentation intermédiaire. Les deux modèles peuvent alors s'utiliser sur n'importe quel type de texte à condition que la représentation intermédiaire soit détectable.

1 Processus d'indexation

1.1 Étapes du processus

Les modèles de recherche d'information présentés rendent compte de la structure sémantique des documents. Cette structure se détecte à partir du texte à l'aide d'outils linguistiques tels que des analyseurs syntaxiques et à l'aide d'information provenant de ressources sémantiques. Normalement, une analyse linguistique produit un graphe sémantique pour chaque phrase d'un document. D'après la théorie Sens Texte, le graphe de chaque phrase représente le sens exprimé par une phrase. La recherche d'information s'intéresse pour sa part au thème de la globalité du document. Elle nécessite de transposer l'ensemble des sens du document en une représentation du thème. Nous émettons ici l'hypothèse que le thème d'un document s'exprime par la synthèse de l'ensemble des représentations du sens (i.e. graphes) de chaque phrase d'un document.

La création de la représentation finale d'un document forme donc un processus en deux étapes. La première produit une représentation du sens pour chacune des phrases, nous l'appelons 'représentation intermédiaire' ; la seconde étape construit la modélisation finale à partir des représentations de chaque phrase (représentation de phrase) pour refléter la thématique globale du document. De nombreux travaux de recherche d'information textuelle basés sur des structures complexes n'utilisent qu'une seule étape. Ils utilisent seulement des représentations de phrase et la correspondance entre les documents et les requêtes se résout uniquement par des correspondances entre les phrases des documents et celles des requêtes. Cette approche nous semble insuffisante et nous mettons en avant la nécessité de construire une représentation globale du document.

Nous avons proposé deux modèles pour la représentation du document. La création de ces deux modèles s'effectue à partir de la représentation intermédiaire d'un document. Comme nous l'avons présenté dans la partie précédente, ces modèles se basent sur deux visions différentes de l'utilisation des supports de vocabulaires. Ces deux modèles se différencient par leur transformation de la représentation intermédiaire.

- Le modèle local (cf. figure 42) représente le contenu des documents. Le graphe final constitue le regroupement de l'information contenue dans chaque représentation de phrase du document. Ce graphe, représentant le thème, concatène l'ensemble des graphes de phrases

du document. Des informations complémentaires à ce graphe, notamment sur son contexte, sont intégrées à la représentation par l'intermédiaire des pondérations qui représentent par exemple l'importance d'un élément du graphe au sein de la collection. Dans ce modèle, le *thème* d'un document s'obtient par la concaténation des *sens* de chaque phrase du document.

- Le modèle global (cf. figure 42) se base sur la création d'un modèle statistique de l'ensemble des représentations de phrase. La représentation du document consiste en un modèle statistique de graphe élaboré à l'aide des représentations de phrase et de la collection. Cette approche extrait le thème du document sous la forme d'un modèle des représentations du *sens* des phrases. Elle distille un condensé de ce qu'exprime l'auteur à travers l'ensemble des sens de ses phrases.

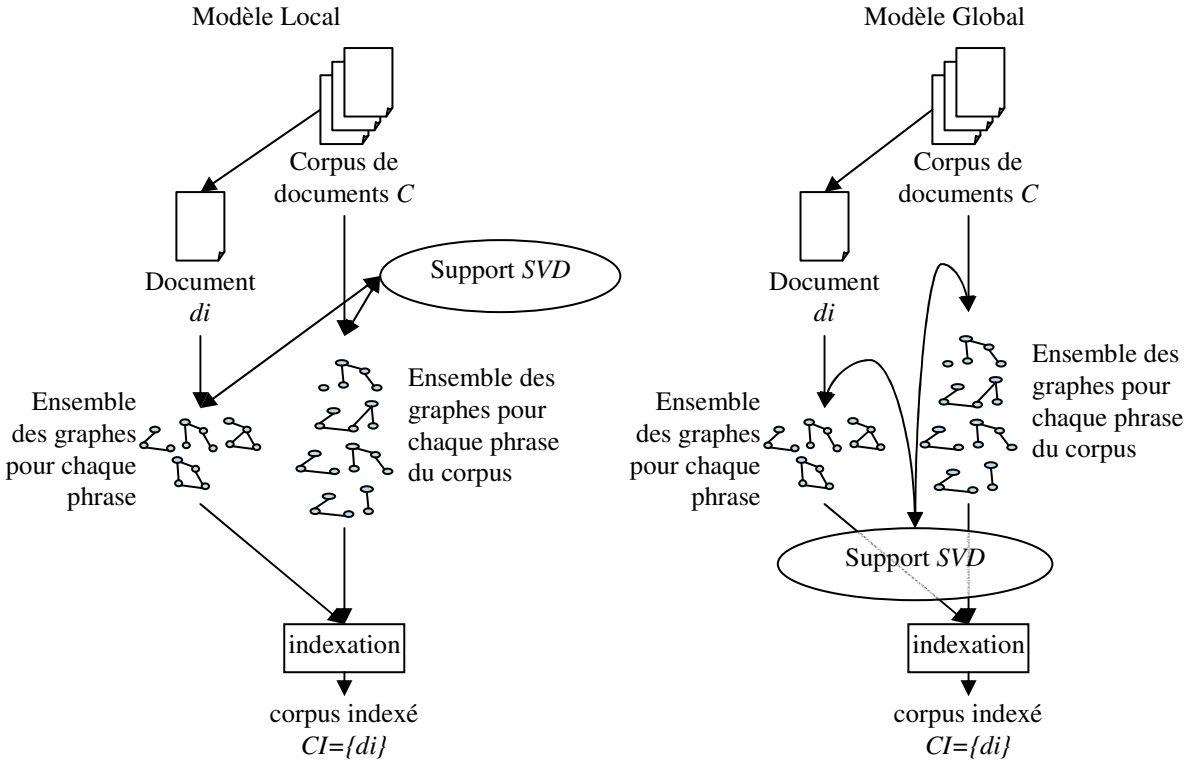


Figure 42 Étapes de la création des deux modèles

1.2 Décomposition de la fonction d'indexation

Dans un premier temps, nous créons la représentation intermédiaire du document basée sur la représentation de chaque phrase par un graphe. Il utilise ensuite ces graphes pour obtenir la représentation finale du document.

Soit un document *d* composé d'un ensemble de *nph* phrases *ph* :

$$d = \{ph\} \text{ avec } |d| = nph$$

On définit alors di_{int} la représentation intermédiaire d'un document composé de *np* représentations de phrases *phi* :

$$di_{int} = \{phi\} \text{ avec } |di_{int}| = nph$$

La fonction d'indexation *ind* permettant l'obtention des représentations de chacun des modèles se décompose alors en :

- Ind^{ph}, une fonction qui lie une phrase *ph* avec sa représentation sous forme de graphe *phi*

$$ind^{ph} : ph \mapsto phi$$

- Ind^{mod}, une fonction qui, pour l'ensemble des représentations de phrases *phi* du document établit le lien avec la représentation définie par le modèle de document.

$$ind^{mod} : \{phi\} \mapsto di$$

La représentation finale *di* du document *d* s'obtient alors par :

$$\begin{aligned} di &= ind(d) \\ &= ind^{mod}(di_{int}) \\ &= ind^{mod}(\{ind^{ph}(ph) | ph \in d\}) \end{aligned}$$

La figure 43 représente cette indexation.

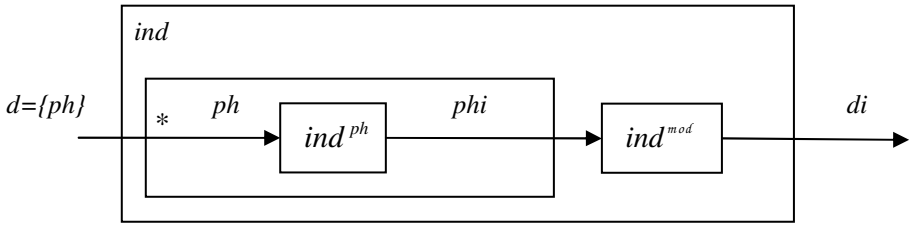


Figure 43 Détails de l'enchaînement des fonctions d'indexation

Nous présentons, par la suite, à l'aide des éléments du chapitre VI, la modélisation intermédiaire utilisée par les deux modèles de recherche d'information.

2 Représentation intermédiaire

La représentation intermédiaire fournit une représentation pour chaque phrase du document. Extraire une telle représentation est une tâche difficile du fait des difficultés linguistiques, notamment l'ambiguïté.

Nous proposons d'intégrer au sein de la représentation de phrase des informations sur le processus d'extraction. Nous représentons cette information à l'aide d'un score de confiance qui représente les erreurs de traitement de la langue et les ambiguïtés incomplètement résolues lors de la création de la représentation de phrase. Normalement, les fonctions qui extraient les éléments à partir du texte doivent procurer ces scores de confiance. Cependant, la majorité des traitements actuels de la langue ne fournissent pas de tels scores. Compte tenu de ce fait, obtenir de tels scores nécessite l'utilisation de méthodes complémentaires que nous présentons dans le chapitre suivant.

Nous proposons de modéliser ces représentations de phrase à l'aide d'un support de vocabulaires et nous fournissons le processus de leur indexation.

2.1 Modèle des représentations de phrase

Nous modélisons les représentations de phrase en utilisant, comme pour les modèles de document, un support de vocabulaires. Le modèle des représentations de phrase est défini par le support de vocabulaires *SVP*. Ce support de vocabulaires se base sur deux vocabulaires, l'un représente les concepts pondérés des phrases, l'autre les relations étiquetées pondérées de la phrase :

$$SVP = \left(V_{conceptsPh}, V_{relationsPh}\right) \text{ avec } nv=2$$

Ces deux vocabulaires pondérés s'établissent à partir du vocabulaire complexe $V_{concepts}$, représentant les concepts, et du vocabulaire complexe $V_{relations}$ représentant les relations, nous posons alors :

- $V_{conceptsPh}$, le vocabulaire des concepts pondérés utilisé pour le modèle de phrase.

  $$V_{conceptsPh} = V_{concepts} \times P^{np_{conceptPh}} \text{ avec } nt=2 \text{ et } np_{conceptPh}=2$$

  Le vocabulaire $V_{conceptsPh}$ associe deux pondérations au vocabulaire $V_{concepts}$. La première représente un score de confiance dans le processus de détection de chaque nœud concept de la phrase. La deuxième donne la fréquence de détection de ce concept au sein de la phrase, nécessaire au calcul de l'importance de ce terme dans le document.

- $V_{relationPh}$, l'ensemble des relations pondérées utilisées pour le modèle de phrase.

  $$V_{relationsPh} = V_{relations} \times P^{np_{relationPh}} \text{ avec } nt=5 \text{ et } np_{relationPh}=2$$

  Le vocabulaire $V_{relationsPh}$ associe deux pondérations au vocabulaire $V_{relations}$. La première représente la confiance dans la détection de la relation au niveau de la phrase, la deuxième représente sa fréquence de détection au sein de la phrase.

2.2 Représentation d'une phrase

a) Formalisme

La représentation *phi* d'une phrase se forme à partir de 2 ensembles :

$$phi = \left(DV^{ph}_{concepts}, DV^{ph}_{relations}\right) \text{ avec :}$$

- $DV^{ph}_{concepts}$ l'ensemble des **concepts** d'une phrase

  $$DV^{ph}_{concepts} \subseteq V_{conceptsPh} \text{ avec } uv_{concept} = \left(uv'_{concept}, p_{confiancePh}, p_{frequence}\right) \in DV^{Ph}_{concepts}$$

où $p_{confiancePh}$ dénote le score de confiance dans la phrase et $p_{frequence}$ dénote le nombre de détections du concept dans la phrase

- $DV^{ph}_{relations}$ l'ensemble des **relations étiquetées** d'une phrase

  $$DV^{ph}_{relations} \subseteq V_{relationsPh} \text{ avec } uv_{relation} = \left(uv'_{relation}, p_{confiancePh}, p_{frequence}\right) \in DV^{Ph}_{relations}$$

où $p_{confiancePh}$ dénote le score de confiance dans la phrase et $p_{frequence}$ constitue le nombre de détections du concept dans la phrase.

b) Contraintes

Dans ce modèle, les sous-ensembles $DV^{ph}_{concepts}$ et $DV^{ph}_{relations}$ représentent les éléments qui apparaissent dans les phrases. Nous émettons trois contraintes sur ces ensembles :

- Unicité des concepts : un concept n'apparaît qu'une seule fois dans une représentation de phrase.
- Unicité des relations : une relation n'apparaît qu'une seule fois dans une représentation de phrase.
- Composition des relations : une relation étiquetée de la représentation de la phrase se forme uniquement de concepts apparaissant dans cette représentation.

Nous présentons un graphe qui correspond à une phrase dans la figure 44, dans un formalisme dérivé des graphes conceptuels.

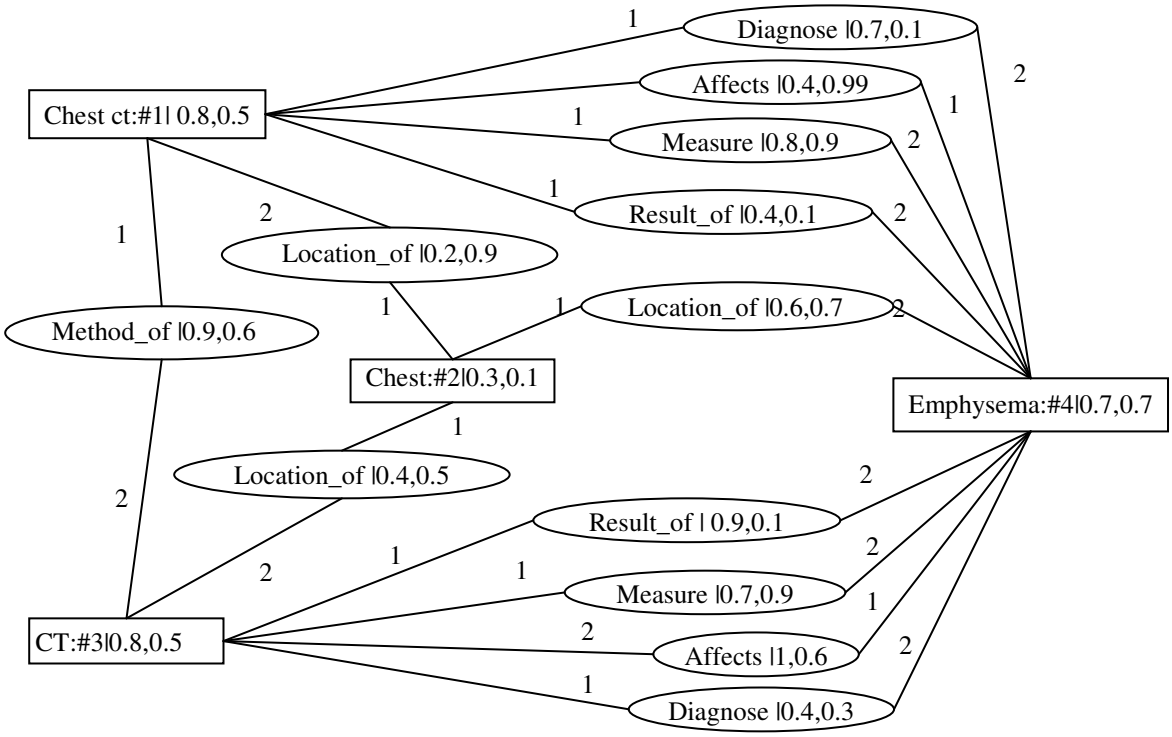


Figure 44 Représentation de la phrase 'Show me chest CT images with emphysema'

2.3 Indexation des représentations de phrase

L'utilisation de nos modèles dans un processus automatique nécessite la définition des méthodes d'indexation des représentations de phrase. Si les ressources externes définissent le support de types fondamental à la création du modèle de recherche d'information, elles doivent fournir suffisamment d'informations pour permettre de lier les éléments de la base de connaissance au texte.

Le système peut utiliser des outils de traitement de la langue fournissant directement les représentations pour chaque phrase. Cependant, si de nombreux outils sont disponibles pour extraire des représentations syntaxiques des phrases, peu d'outils sont disponibles ou suffisamment efficaces pour générer des représentations sémantiques des phrases. Nous proposons ici des méthodes qui

utilisent les informations contenues dans des ressources sémantiques pour générer ces représentations. Ces ressources doivent définir des concepts et des relations sémantiques entre les concepts

Pour créer la représentation à partir d'une phrase, nous détectons de prime abord les concepts dans la phrase, puis nous détectons les relations entre ces concepts. La fonction d'indexation des phrases Ind^{ph} se décompose alors en deux fonctions :

- $Ind^{ph}_{concept}(ph)$ une fonction qui, à partir d'une phrase, fournit l'ensemble des nœuds concepts qui apparaissent dans cette phrase.

$$Ind^{ph}_{concept}(ph) = DV^{ph}_{concepts}$$

- $Ind^{ph}_{relation}\left(ph, DV^{ph}_{concepts}\right)$ une fonction qui, à partir d'une phrase et d'un ensemble de concepts, fournit la représentation de la phrase contenant les concepts et les relations entre ces concepts.

$$Ind^{ph}_{relation}\left(ph, DV^{ph}_{concepts}\right) = \left(DV^{ph}_{concepts}, DV^{ph}_{relations}\right)$$

La fonction d'indexation Ind^{ph} permettant d'obtenir la représentation de phrase *phi* à partir d'une phrase *ph*, s'écrit :

$$Ind^{ph}(ph) = \left(DV^{ph}_{concepts}, DV^{ph}_{relations}\right) = Ind^{ph}_{relation}\left(Ind^{ph}_{concept}(ph), ph\right)$$

La figure 45 résume cette fonction d'indexation qui décrit le processus de génération d'une représentation de phrase.

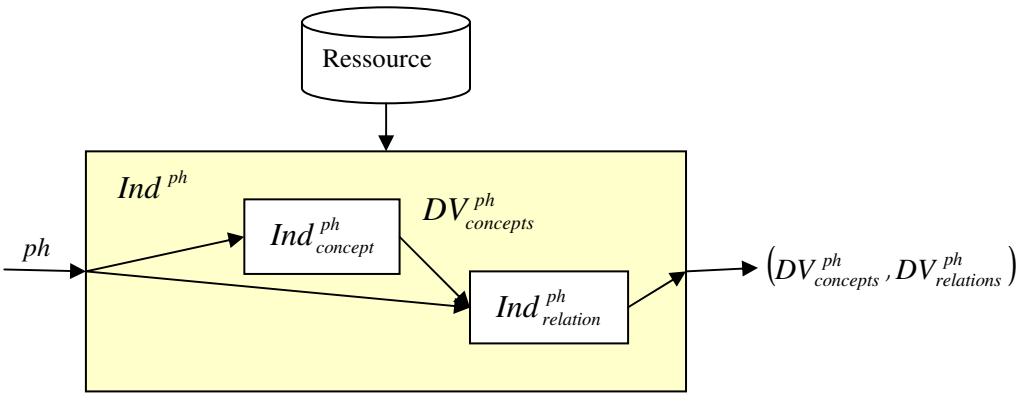


Figure 45 Génération d'une représentation de phrase

2.4 Conclusion

Sur le texte, la construction des représentations des documents constitue un processus en deux étapes. La première définit la création d'une représentation intermédiaire du document qui fournit un ensemble de représentations de phrase. Ces représentations de phrase peuvent se modéliser à l'aide d'un support de vocabulaires *SVP*. Cette représentation s'obtient à l'aide de la fonction d'indexation intermédiaire I_{ph} qui permet de créer une représentation pour chaque phrase du document. La deuxième étape s'intéresse au regroupement de l'ensemble de ces graphes au sein d'une structure unique qui représente le document. Cette étape diffère pour les deux modèles proposés. Par la suite,

nous définissons les caractéristiques des deux modèles pour le texte et nous donnons les méthodes de création de la représentation des documents pour les deux modèles.

3 Modèle Local *ML*

Nous obtenons le modèle *ML* par concaténation de la représentation de chaque phrase du document en une représentation unique synthétisant les informations du document. Le modèle local, tel que présenté dans le chapitre VII, ne définit pas le nombre de pondérations utilisées par les deux supports de vocabulaires SVD_{ML} et SVQ_{ML}. De même, l'instanciation de ce modèle nécessite de définir la fonction qui permet le calcul du degré de pertinence. Nous présentons ici nos choix pour l'utilisation de ce modèle sur le texte. Nous exposons ensuite la fonction d'indexation qui permet la concaténation des représentations de phrase, ainsi que la fonction d'interprétation de la requête.

3.1 Application du modèle

3.1.1 Choix des poids

Nous utilisons deux pondérations pour les concepts et les relations du modèle *ML,* que ce soit pour les requêtes ou pour les documents :

$$np_{conceptDoc}=2,\ np_{relationDoc}=2,\ np_{conceptReq}=2 \text{ et } np_{relationReq}=2$$

- Un concept se représente alors par :

$$uv_{concept} = \left(uv'_{concept}, p_{confiance}, p_{ri}\right) \in V^{ML}_{conceptsDoc} = V_{concepts} \times P^2$$

- Une relation se représente alors par :

$$uv_{relation} = \left(uv'_{relation}, p_{confiance}, p_{ri}\right) \in V^{ML}_{relationsDoc} = V_{relations} \times P^2$$

La première pondération ($p_{confiance}$) représente la confiance dans le processus d'extraction d'un élément au sein de la représentation finale du document. Ce poids dépend du processus d'extraction des éléments : l'importance de ce poids dépend de la fiabilité du processus qui a amené à la détection de l'élément au sein de la représentation du document. Ce poids permet d'intégrer au sein de la représentation finale du document une partie des erreurs et des ambiguïtés du processus d'extraction des graphes de phrases $P_{confiancePh}$.

La deuxième pondération (p_{ri}) représente l'importance de l'élément pour la recherche d'information. Cette pondération se calcule typiquement à l'aide de variations du *tf.idf*. Elle détermine l'importance des éléments constitutifs du graphe final du document. Cette pondération permet d'intégrer au sein des documents des informations sur la collection, notamment avec l'utilisation de l'*idf*. Ces informations ne se calculent cependant que pour les éléments qui apparaissent dans le document.

3.1.2 Fonction de correspondance

Dans ce modèle, la fonction de correspondance repose sur le calcul du degré de correspondance présenté dans la section 3.4 du chapitre VII. Ce calcul rend compte de l'importance de la correspondance de deux concepts pour la recherche d'information. Dans notre instanciation, la correspondance est forte quand les deux concepts (ou relations) qui correspondent sont importants dans le document (p_{ri} fort) et sont extraits de façon certaine ($p_{confiance}$ fort). Au contraire, quand un des deux éléments en correspondance est peu important (p_{ri} faible) ou incertain ($p_{confiance}$ faible) la

correspondance doit en être réduite proportionnellement. Pour ces raisons, nous implémentons le calcul du degré de correspondance par la multiplication des quatre poids associés aux deux concepts (ou relations) :

$$f_{concepts}\left(p^q_{concept}, p^d_{concept}\right) = f_{concepts}\left(p^q_{confiance}, p^q_{ri}, p^d_{confiance}, p^d_{ri}\right) = p^q_{confiance} \times p^q_{ri} \times p^d_{confiance} \times p^d_{ri},$$

$$f_{relations}\left(p^q_{relation}, p^d_{relation}\right) = f_{relations}\left(p^q_{confiance}, p^q_{ri}, p^d_{confiance}, p^d_{ri}\right) = p^q_{confiance} \times p^q_{ri} \times p^d_{confiance} \times p^d_{ri}$$

3.2 Fonction d'indexation

Ce modèle ne représente que les informations disponibles dans le document. Il se base sur la concaténation de l'ensemble des représentations de phrase. Si un même concept (ou relation) apparaît dans deux représentations de phrase du document, ces deux concepts (ou relations) sont considérés comme identiques, ils n'apparaissent qu'une seule fois dans la représentation du document. Le nombre d'apparitions d'un élément du graphe est pris en compte par son poids.

Un document *di* s'obtient à partir de la représentation intermédiaire du document di_{int}. Cette représentation se compose d'un ensemble de représentations de phrase *phi*, elles-mêmes composées d'un ensemble de concepts et d'un ensemble de relations :

$$\begin{aligned} di &= \left(DV^{ML}_{concepts}, DV^{ML}_{relations}\right) \\ &= ind^{mod\ ML}\left(di_{int}\right) \\ &= ind^{mod\ ML}\left(\{phi\}\right) \\ &= ind^{mod\ ML}\left(\left\{\left(DV^{ph}_{concepts}, DV^{ph}_{relations}\right)\right\}\right) \end{aligned}$$

3.2.1 Concepts

L'ensemble des concepts $DV^{ML}_{concepts}$ est formé par la concaténation des vocabulaires non pondérés des concepts de chaque représentation de phrase *phi* de la représentation intermédiaire di_{int}.

$$f_{np}\left(DV^{ML}_{concepts}\right) = \bigcup_{phi} f_{np}\left(DV^{ph}_{concepts}\right)$$

Le poids de recherche d'information p_{ri} d'un concept s'obtient par le calcul du *tf.idf* ou d'une variation de celui-ci. Ce calcul se base sur la fréquence des concepts au sein des phrases du document et sur la fréquence des concepts dans les documents.

$$p_{ri} = tf.idf$$

Le poids de confiance $p_{confiance}$ d'un concept $uv_{concept}$ se calcule par la somme des poids de confiance de ce concept pour chaque représentation de phrase. Un concept extrait depuis plusieurs phrases d'un même document à donc un score plus élevé qu'un concept extrait depuis une seule phrase. Le calcul du score de confiance se calcule selon :

$$p_{confiance} = \sum_{phi}\left(p_{confiance}\right)$$

3.2.2 Relations

Comme pour les concepts, l'ensemble des relations $DV^{ML}_{relations}$ est formé par la concaténation des vocabulaires non pondérés des relations de chaque représentation de phrase *phi* de la représentation intermédiaire.

$$f_{np}\left(DV^{ML}_{relations}\right)=\bigcup_{phi} f_{np}\left(DV^{ph}_{relations}\right)$$

De même que pour les concepts, le poids de recherche d'information p_{ri} d'une relation $uv_{relation}$ s'obtient par le calcul d'un *tf.idf* basé sur les fréquences stockées dans les représentations de phrase. Le poids de confiance $p_{confiance}$ d'une relation $uv_{relation}$ est calculé par la somme des poids de confiance de cette relation pour chaque représentation de phrase du document :

$$p_{ri}=tf.idf \text{ et } p_{confiance}=\sum_{phi}\left(p_{confiance}\right)$$

3.3 Interprétation de la requête

Dans ce modèle, le besoin d'un utilisateur est exprimé sous la forme d'une phrase. Sa formulation dans le système de recherche d'information correspond donc à la représentation d'une phrase détectée à partir de la requête.

$$q=Inter(b)=Ind^{ph}(ph)$$

où *ph* est une phrase qui exprime le besoin *b* de l'utilisateur.

3.4 Conclusion

Nous avons présenté les différentes caractéristiques du modèle *ML* pour son application au texte. Pour ce modèle, la représentation du document consiste en une fusion de l'ensemble des représentations de phrase de ce document. Cette représentation forme un graphe unique qui contient deux pondérations pour chaque élément qui le constitue. L'application de ce modèle à un certain domaine ne nécessite plus que la définition des fonctions qui permettent la création de la représentation intermédiaire du document, ce que nous présentons dans le chapitre suivant.

4 Modèle Global *MG*

Le modèle *MG* se base sur une modélisation du document. Pour définir l'application de ce modèle au document textuel, nous utilisons le processus génératif qui sous-tend la modélisation. Nous présentons dans un premier temps l'utilisation de ce modèle sur la représentation intermédiaire des documents, puis nous détaillons le calcul des estimations de ce modèle.

4.1 Instanciation du Modèle

Dans ce cas, nous utilisons une pondération pour les concepts, les couples et les relations du modèle de document tandis que le modèle de requête n'utilise pas de pondérations :

$$np_{conceptDoc}=1,\ np_{relationDoc}=1, \text{ et } np_{relationDoc}=1$$

Ce modèle pondère tous les éléments du vocabulaire, nous créons en effet un modèle de graphe pour chaque document. Nous reprenons le processus génératif, présenté dans le chapitre VII, qui modélise directement des graphes. La probabilité d'un graphe de requête est obtenue par :

$$P\left(G_Q \middle| M_D^g\right) = P\left(C_Q \middle| M_D^g\right) \times P\left(R_Q \middle| C_Q, M_D^g\right)$$

avec $P\left(R_Q \middle| M_D^g\right) = \prod_{R_Q} P\left(type(r) = \{tr\} \middle| c1, c2, M_D^g\right)$

Dans un cas nous estimons directement la probabilité $P\left(type(r) = \{tr\} \middle| c1, c2, M_D^g\right)$, dans l'autre nous la décomposons en deux probabilités :

$$P\left(type(r) = tr \middle| c1, c2, M_D^g\right) = P\left(couple(c1, c2) = 1 \middle| c1, c2, M_D^g\right) \times P\left(type(r) = tr \middle| couple(c1, c2) = 1, M_D^g\right)$$

4.2 Fonction d'indexation

Ce modèle représente tous les éléments du vocabulaire, il n'effectue pas de sélection entre les éléments qui apparaissent dans le document et ceux qui n'y apparaissent pas. La distinction entre les représentations s'effectue par les estimations qui donnent des probabilités plus importantes aux éléments qui apparaissent dans le document par rapport aux autres. Nous détaillons dans les sections suivantes les calculs utilisés pour élaborer les estimations.

Un document indexé *di* se forme à partir de la représentation intermédiaire di_{int} constituée d'un ensemble de représentations de phrase *phi* :

$$\begin{aligned} di &= \left(DV_{concepts}^{ML}, DV_{couples}^{ML}, DV_{relations}^{ML}\right) \\ &= ind^{mod\ MG}\left(di_{int}\right) \\ &= ind^{mod\ MG}\left(\{phi\}\right) \\ &= ind^{mod\ MG}\left(\left\{\left(DV_{concepts}^{ph}, DV_{relations}^{ph}\right)\right\}\right) \end{aligned}$$

4.2.1 Concepts

Nous proposons d'estimer la probabilité des concepts par la probabilité suivante :

$$P\left(uv'_{concept} \middle| M_D^g\right) = \left(1 - \lambda_{concept}\right) \frac{D\left(uv'_{concept}\right)}{D(*)} + \lambda_{concept} \frac{C\left(uv'_{concept}\right)}{C(*)}$$

où $D\left(uv'_{concept}\right)$ représente la fréquence de $uv'_{concept}$ au sein de la représentation intermédiaire *di'* et *D(*)* dénote la somme des fréquences de l'ensemble des concepts de la représentation intermédiaire du document. $C\left(uv'_{concept}\right)$ et *C(*)* représentent des quantités équivalentes mais calculées sur l'ensemble de la collection.

Cette probabilité correspond à la probabilité unigramme, habituellement utilisée dans les modèles de langue, que nous appliquons ici aux concepts. Le calcul de cette probabilité s'effectue par un simple lissage de Jelinek-Mercer sur la collection. Ce lissage fait intervenir $\lambda_{concept}$ qui doit être estimé sur une base d'apprentissage.

4.2.2 Couples

Dans le cas où le modèle décompose la probabilité des relations, nous estimons la probabilité des couples par la probabilité suivante :

$$P\left(uv'_{couple} \middle| uv'_{concept1}, uv'_{concept2}, M_D^g\right) = \left(1-\lambda_{couple}\right)\frac{D\left(uv'_{couple}\right)}{D\left(uv'_{concept1}, uv'_{concept2}\right)} + \lambda_{couple}\frac{C\left(uv'_{couple}\right)}{C\left(uv'_{concept1}, uv'_{concept2}\right)}$$

où $D\left(uv'_{couple}\right)$ représente le nombre de fois où $uv'_{concept1}$ et $uv'_{concept2}$ forment un couple uv'_{couple} dans une des représentations de phrase *phi* du document. $D\left(uv'_{concept1}, uv'_{concept2}\right)$ dénote le nombre de fois où les deux concepts apparaissent dans le même graphe. $C\left(uv'_{couple}\right)$ et $C\left(uv'_{concept1}, uv'_{concept2}\right)$ correspondent aux quantités équivalentes calculées sur la collection.

Comme pour la probabilité précédente, nous utilisons un lissage de Jelinek-Mercer entre l'estimation de la probabilité calculée sur le document et celle calculée sur le corpus. Ce calcul se rapproche du calcul de la probabilité des bigrammes tel que présenté dans (Srikanth et Srihari, 2002)

4.2.3 Relations

Dans le cas où le système ne décompose pas la probabilité des relations, nous estimons la probabilité des relations sachant les concepts par la probabilité suivante :

$$P\left(uv'_{relation} \middle| uv'_{concept1}, uv'_{concept2}, M_D^g\right) = \left(1-\lambda_{relation}\right)\frac{D\left(uv'_{relation}\right)}{D\left(uv'_{concept1}, uv'_{concept2}\right)} + \lambda_{relation}\frac{C\left(uv'_{relation}\right)}{C\left(uv'_{concept1}, uv'_{concept2}\right)}$$

Où $D\left(uv'_{relation}\right)$ représente le nombre de fois où la relation $uv'_{relation}$ apparaît dans une des représentations de phrase *phi* du document et $C\left(uv'_{relation}\right)$ dénote la quantité équivalente sur la collection.

Dans le cas où le système décompose la probabilité des relations, nous estimons la probabilité des relations sachant les couples par la probabilité suivante :

$$P\left(uv'_{relation} \middle| uv'_{couple} M_D^g\right) = \left(1-\lambda_{relation}\right)\frac{D\left(uv'_{relation}\right)}{D\left(uv'_{couple}\right)} + \lambda_{relation}\frac{C\left(uv'_{relation}\right)}{C\left(uv'_{couple}\right)}$$

4.3 Interprétation de la requête

Dans ce modèle, comme pour le précèdent, le besoin d'un utilisateur s'exprime sous la forme d'une phrase. Une requête du modèle *MG* se représente par un graphe qui correspond à la représentation de phrase sans les pondérations.

$$q = Inter(b) = f_{np}\left(Ind^{ph}(ph)\right)$$

où *ph* dénote l'expression du besoin *b* de l'utilisateur.

4.4 Conclusion

Cette section a présenté les caractéristiques du modèle *MG* pour son application au texte. Dans ce modèle, la représentation du document constitue une modélisation statistique de l'ensemble des représentations de phrase. Nous proposons deux variations du calcul de ce modèle avec les estimations correspondantes. Nous soulignons cependant que ces estimations ne tiennent pas compte des scores de confiance calculés sur les concepts et les relations des phrases. Comme pour le modèle précédent, l'application de ce modèle à un domaine ne nécessite plus que la détection de la représentation intermédiaire.

5 Bilan

Les deux modèles présentés se basent sur une représentation intermédiaire qui représente un document par un ensemble de représentations de phrase.

À partir de cette représentation intermédiaire, l'objectif de ces deux modèles consiste à faire ressortir le thème. Ces modèles explorent des méthodes différentes pour arriver à leur fin. Le modèle local se base sur la concaténation du contenu de chaque phrase du document. Le modèle global se base sur la création d'un modèle de graphe à partir de l'ensemble des représentations de phrase d'un document.

L'intérêt de la décomposition du processus en deux étapes est de fournir des méthodes génériques à plusieurs domaines pour la construction des modèles finaux. Ces méthodes s'appliquent en effet sur n'importe quelle représentation intermédiaire des documents tant que celle-ci respecte le modèle présenté dans ce chapitre.

Dans le chapitre suivant nous proposons un support de type et différentes méthodes qui permettent de créer les représentations de phrase pour le domaine médical. Nous proposons également dans l'annexe B une application dégradée de ces modèles à l'analyse syntaxique des textes, analyses qui fournissent des représentations de phrase sous la forme d'arbres de dépendance. Ces deux applications attestent de la généricité de notre approche.

Chapitre IX Application aux Textes Médicaux

« Le langage est-il l'expression adéquate de toutes les réalités ? » Friedrich Nietzsche (Le livre du philosophe)

La recherche d'information sur des domaines spécifiques prend de plus en plus d'importance. Dans ces domaines, les utilisateurs possèdent des besoins experts. Ils constituent des domaines appropriés au développement de systèmes orientés précision. Le domaine médical n'échappe pas à cette règle et constitue un domaine intéressant pour évaluer des méthodes orientées précision. De plus, de nombreux travaux de recherche portent sur la formalisation de ce domaine à l'aide de ressources sémantiques telles que UMLS (Kleinsorge *et al.*, 2006) ou MESH[13].

Ce chapitre présente les éléments nécessaires à l'application des deux modèles de recherche d'information aux textes médicaux. Cette application passe d'abord par la caractérisation des types qui composent le support de type commun aux modèles. Cette caractérisation est importante car elle fixe le sens final des modèles.

L'application passe ensuite par la création de méthodes permettant l'obtention des représentations de phrases. Ces méthodes servent de base à l'ensemble du processus de construction des modèles, leur efficacité est importante. Sur le texte, obtenir des représentations sémantiques est difficile. Nous testons donc plusieurs méthodes de détection des graphes, et nous proposons l'utilisation de scores de confiance pour témoigner de la performance des processus de détection.

La première partie de ce chapitre présente le support de types utilisé sur les textes médicaux qui se base sur l'une des ressources du domaine : UMLS. Les deux dernières parties présentent les fonctions qui permettent la création des représentations de phrases à l'aide de ce support, la première présente l'extraction des concepts, la deuxième l'extraction des relations.

1 Support de types

Nous sélectionnons dans nos expérimentations une ressource du domaine médical préexistante pour créer le support de types des deux modèles : UMLS. Cette ressource permet de définir des concepts et des relations sémantiques entre ces concepts. Pour que la création des représentations de phrase soit automatisable, cette ressource fournit des éléments nécessaires à la création de ces représentations. Après avoir décrit UMLS, nous présentons les éléments sélectionnés afin de correspondre aux types utilisés dans le support de types des modèles.

[13] http://www.nlm.nih.gov/mesh/

1.1 Unified Medical Language System (UMLS)

UMLS[14] (Kleinsorge *et al.*, 2006) constitue la ressource choisie pour générer nos représentations des documents. UMLS est un méta-thésaurus couvrant le domaine médical. Il résulte de la fusion de nombreuses ressources (plus de 110), thésaurus et terminologies, décrites dans différentes langues. Le *National Library of Medicine* (NLM) réalise et maintient cette ressource. C'est actuellement l'une des ressources les plus importantes en termes de taille et de couverture pour la médecine.

UMLS se divise en trois composants principaux qui forment trois sources de connaissances :

- Le **méta-thésaurus** qui représente la part la plus importante d'UMLS. Il se compose d'environ un million de concepts reliés à 5 millions de termes. La construction de ce thésaurus se base sur la fusion de différentes sources, thésaurus et listes d'autorité couvrant en grande partie le domaine médical. Cette fusion identifie des concepts, reliés entre eux au travers des différents thésaurus. Elle fournit des informations sur leurs expressions textuelles à l'aide de termes dans différentes langues.
- Le **réseau sémantique** qui définit des relations générales de niveau sémantique entre des classes de concepts (cf. figure 46). Les classes de concepts, définies comme des types sémantiques, forment des catégories larges au nombre de 135, telles que '*Disease or Syndrome*'. 54 relations relient ces catégories. Ces dernières ont des sens généraux, par exemple : '*Virus→causes→Disease or Syndrome*'. Une partie de ces relations permet de créer un réseau entre ces types sémantiques.

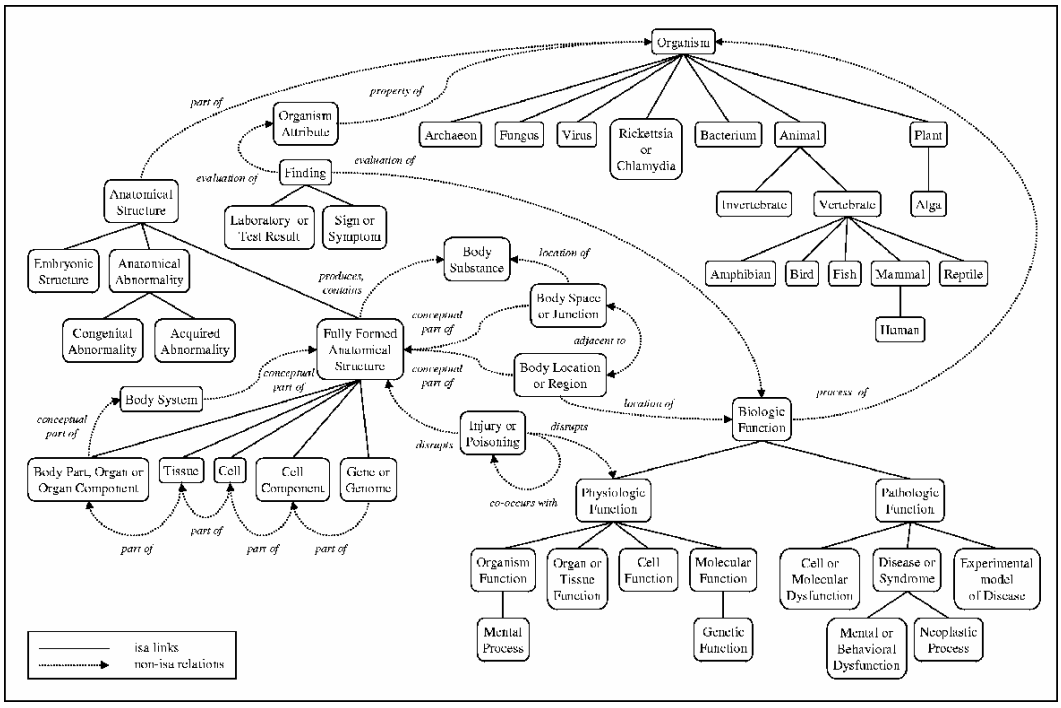


Figure 46 Une partie du réseau sémantique de UMLS[15]

[14] http://umlsks.nlm.nih.govumlsks.nlm.nih.gov/

[15] http://www.nlm.nih.gov/research/umls/documentation.html

- Le **lexique de spécialité** qui contient des informations lexicales et des programmes pour le traitement de la langue. Le but du lexique consiste à résoudre la variation en anglais.

Par sa taille qui permet de couvrir en grande partie les concepts du domaine médical, et par son aspect multilingue, UMLS constitue un bon candidat pour la production de graphes à partir de textes multilingues du domaine médical. UMLS n'est cependant pas une ontologie car il ne contient pas de description formelle des concepts, mais il fournit de nombreux termes et leurs variations pour chacun des concepts. UMLS constitue une ressource intéressante pour évaluer des indexations conceptuelles à grande échelle.

Nous notons qu'une telle ressource ne peut pas contenter tous les praticiens d'un domaine de spécialité, car toute classification contraint la réalité dans un point de vue toujours discutable. Cependant, UMLS fournit un point de vue à un instant donné, et même si ce point de vue ne constitue pas l'unanimité, il peut être utilisé, particulièrement en recherche d'information. De plus, comme UMLS constitue un regroupement de thésaurus, il peut aussi se percevoir par différents points de vue. Cela se réalise en choisissant les thésaurus les plus pertinents par rapport à une tâche donnée, ce qui est conseillé par les auteurs d'UMLS (cf. manuel UMLS).

Nous sélectionnons ainsi dans UMLS les éléments nécessaires à la construction de nos graphes. Ces éléments forment le support de types de notre modèle. Les sections suivantes présentent donc les trois types qui composent ce support.

1.2 Concepts $T_{concepts}$

Dans UMLS les concepts correspondent à des abstractions non définies formellement, de plus UMLS les relie à plusieurs de leurs expressions textuelles (termes). En recherche d'information, l'intérêt premier lors de la construction du méta-thésaurus découle de l'étape qui regroupe les abstractions identiques correspondant à chaque expression (terme) de chaque vocabulaire source. Dans un certain sens, UMLS crée des clusters de termes ayant un sens commun et définit ces clusters comme des concepts.

Chaque concept du méta-thésaurus possède un identifiant unique (CUI ; Concept Unique Identifier) auquel se lie un certain nombre de termes (LUI : Term Unique Identifier). Ces LUIs se relient eux aussi à un certain nombre de chaînes de caractères (SUI : String Unique Identifier). Enfin ces chaînes peuvent être définies dans plusieurs vocabulaires sources (AUI Atome Unique Identifier). Le tableau 15 donne un exemple de cet enchaînement.

Concepts (CUI)	Termes (LUIs)	Chaînes (SUIs)	Atomes (AUIs)
C0004238 Atrial Fibrillation (préféré) Atrial Fibrillations Auricular Fibrillation Auricular Fibrillations	**L0004238** Atrial Fibrillation (préféré) Atrial Fibrillations	**S0016668** Atrial Fibrillation (préféré)	**A0027665** Atrial Fibrillation (de MSH) **A0027667** Atrial Fibrillation (de PSY)
		S0016669 Atrial Fibrillations	**A0027668** Atrial Fibrillations (de MSH)
	L0004327 (synonyme) Auricular Fibrillation Auricular Fibrillations	**S0016899** Auricular Fibrillation (préféré)	**A0027930** Auricular Fibrillation (de PSY)
		S0016900 (pluriel) Auricular Fibrillations	**A0027932** Auricular Fibrillations (de MSH)

Tableau 15 Exemple de concepts et d'expressions associées (cf. manuel UMLS)

- Un **concept** représente le sens d'un cluster de termes.
- Un **terme** représente une expression d'un concept sous forme de termes. Pour un concept, l'ensemble des termes de ce concept représente les variations d'expression de ce concept.
- Une **chaîne** représente une variation d'un terme au niveau de la chaîne de caractères. L'ensemble des chaînes correspondant à un terme représente ses variations au niveau des caractères, que ce soit lexical ou au niveau de la casse.
- Un **atome** constitue le plus petit élément dans la structure. Il représente une chaîne de caractères provenant d'une source précise. L'ensemble des atomes d'une chaîne forme l'ensemble des sources où cette chaîne est décrite.

UMLS ne définit pas toutes les chaînes de caractères possibles pour un concept. C'est une vue partielle du lien concepts-termes. Cependant le grand nombre de chaînes définies constitue un plus pour faire la correspondance entre le texte et les concepts qui apparaissent dans ce texte.

Les termes qui correspondent aux concepts sont exprimés dans différentes langues, cependant l'anglais reste largement majoritaire dans UMLS, nous donnons dans le tableau 16 les détails sur le nombre de termes exprimés pour quelques-unes des langues du méta-thésaurus de la version 2007 de UMLS.

	Nombre de termes	Pourcentage
Total	6 772 669	
Anglais	4 297 431	63,45%
Français	156 404	2,31%
Allemand	168 186	2,48%
espagnol	1 322 167	19,52%

Tableau 16 Détail des langues de UMLS version 2007AA

En plus du lien entre concepts et chaînes textuelles, chaque concept d'UMLS se relie à une ou plusieurs classes sémantiques (définies dans le réseau sémantique). Ces liens attribuent à chaque concept un ou plusieurs types sémantiques (TUI). Le tableau 17 décrit deux exemples de concepts associés à leurs types sémantiques.

Concept	Type Sémantique
C0004238(Atrial Fibrillation)	T046(Pathologic Function)
C0817096(Chest)	T029(Body Location or Region)

Tableau 17 Lien concepts - types sémantiques

Le Modèle Local peut utiliser un ordre sur les concepts, cependant obtenir des ordres partiels cohérents dans une ressource est généralement difficile. Les relations proposées ici dans le méta-thésaurus d'UMLS ne permettent pas de définir un treillis raisonnable. Ces relations proviennent en effet de différentes sources ; si elles forment une arborescence correcte sur le thésaurus d'origine, elles ne forment pas, même regroupées, un treillis sur UMLS. Par conséquent nous n'utilisons pas le treillis dans nos modèles.

Les concepts tels que décrits dans UMLS se détectent grâce à leurs formes textuelles. Nous les utilisons donc pour décrire le type concept. Le type concept $T_{concepts}$ utilisé par l'application des modèles aux textes médicaux est défini par :

$$T_{concepts}=\{CUI\ défini\ dans\ UMLS\}$$

1.3 Relations

Le méta-thésaurus se compose de relations provenant de tous les thésaurus. Ces relations s'avèrent très variées et peu organisées. Elles proviennent de chaque thésaurus et ne subissent aucune normalisation au moment du regroupement des thésaurus ; par conséquent elles sont difficiles à utiliser. Une solution consiste à restreindre le thésaurus à une source de vocabulaire précise (par exemple Mesh), cependant cela limite les concepts qui peuvent être reliés par ces relations.

Les concepts peuvent aussi se relier à partir d'informations obtenues sur le réseau sémantique. Le réseau sémantique définit un certain nombre de relations générales entre les classes sémantiques. Une relation sémantique représente un lien possible entre deux concepts si ces deux concepts correspondent aux catégories sémantiques liées par la relation.

Nous utilisons les relations sémantiques définies dans le réseau sémantique d'UMLS. Il y a 54 relations sémantiques. Ce faible chiffre est un avantage en RI car il limite la variabilité des relations. De plus, le réseau sémantique définit des relations assez générales et donc simples à manipuler.

Le type concept $T_{concepts}$ utilisé par les applications des modèles aux textes médicaux se définit alors par :

$$T_{relations}=\{relations\ du\ réseau\ sémantique\}$$

1.4 Référents

Nous émettons ici l'hypothèse simplificatrice que, dans un document textuel, tous les concepts de même type possèdent le même référent. Si un concept d'un certain type apparaît dans deux phrases distinctes alors ce concept reste le même dans les deux phrases, il possède le même nom et le même référent. Le référent constitue simplement une numérotation dépendante du type de concept.

$$T_{référents}=\{numéro\}$$

Notre modèle ne tient pas compte du référencement multiple même si un tel référencement peut s'avérer utile sur des domaines techniques (Kefi-Khelif, 2006). Dans le domaine médical, les concepts utilisés se désignent rarement par un référent. Si dans un document textuel un concept précis apparaît plusieurs fois, nous pensons raisonnable de considérer que ce concept est, soit relié au référent générique, soit désigne à chaque fois le même élément. Enfin, sur des textes médicaux qui ne désignent pas explicitement les référents, leur détection reste difficile et nécessite des traitements de la langue complexes, contrairement aux documents techniques présentés dans (Kefi-Khelif, 2006) où les textes désignent explicitement les référents.

Par conséquent, nous ignorons le référent d'un concept dans un document. Nous considérons par la suite l'égalité de deux concepts si et seulement si ils possèdent le même type de concept, que ce soit dans le même document ou entre le document et la requête.

1.5 Synthèse

Nous utilisons UMLS pour extraire des graphes. Les nœuds de ces graphes se forment à l'aide des concepts tels que définis dans UMLS. Les relations sémantiques qui relient ces nœuds se forment des relations définies dans le réseau sémantique d'UMLS. UMLS fournit des informations qui permettent la génération des représentations de phrase, notamment des termes permettant la détection des concepts dans le texte. Le support des deux modèles appliqués aux textes médicaux se résume par :

$$ST_{UMLS} = (T_{concepts}, T_{relations}, T_{référents})$$

avec $T_{concepts}$={*CUI défini dans UMLS*}, $T_{relations}$={*relations du réseau sémantique*},$T_{référents}$={*numéro*}

2 Détection des concepts

Cette section présente des méthodes qui permettent d'implémenter la fonction $ind_{concept}^{ph}$ qui fournit l'ensemble des concepts détectés dans une phrase. La littérature propose de nombreuses méthodes pour détecter les concepts. Certaines utilisent des traitements de la langue pour détecter les mots et les syntagmes correspondant aux concepts (Aronson, 2001). D'autres, plus combinatoires (Hersh et Donohoe, 1998)(Nadkarni *et al.*, 2001)(Zou *et al.*, 2003), se basent sur la cooccurrence dans les phrases, des mots composant les termes.

Nous proposons ici deux méthodes pour détecter les concepts. La première utilise l'outil MetaMap qui ne traite que les textes en anglais, ce qui empêche de tester des approches multilingues. Nous avons donc développé une deuxième méthode qui se base sur un étiquetage morphosyntaxique du texte et sur une détection des concepts à l'aide de correspondances de termes. Cette seconde méthode est applicable sur différentes langues à condition de disposer d'un analyseur morphosyntaxique de cette langue. Nous proposons deux variations de cette méthode basées sur l'utilisation de deux analyseurs morphosyntaxiques différents : le premier TreeTagger permet d'analyser des textes en français, en anglais et en allemand, le deuxième MiniPar permet seulement de traiter des textes en anglais mais fournit une analyse de dépendance que nous utilisons pour calculer le score de confiance sur les relations, tel que nous le présentons dans la section 3.1 de ce chapitre.

Si nous proposons trois extractions pour des raisons différentes, nous tirons partie des ces différentes extractions. Cela nous permet de comparer différentes approches sur nos modèles, voire de combiner nos modèles, ce que nous effectuons dans les expérimentations de la partie 5.

2.1 Méthode Metamap

La première méthode d'extraction des concepts utilise l'analyseur MetaMap (Aronson, 2001). MetaMap désigne un outil de détection de vocabulaire médical contrôlé (à l'aide d'UMLS) à partir de documents médicaux. Cet analyseur se base sur une décomposition du texte en syntagmes nominaux.

Pour chaque syntagme détecté dans le texte, l'outil génère un ensemble de variantes possibles pour ce syntagme : d'une part, en ne prenant qu'une partie des mots constituant le syntagme, d'autre part, en utilisant des variantes : écriture, acronymes, dérivations lexicales, etc.

L'outil établit ensuite la liste des termes du méta-thésaurus qui contiennent l'une de ces variantes, cette liste s'appelle la liste des candidats. Pour chaque candidat, l'outil calcule un score en fonction de différents paramètres.

MetaMap établit ensuite une liste d'*ensembles candidats* constituée de candidats portant sur des parties disjointes du syntagme nominal ; un ensemble s'appelle une correspondance. MetaMap calcule

un nouveau score pour ces combinaisons de candidats. Au final, les correspondances avec les plus forts scores contiennent les *meilleurs concepts*.

MetaMap fournit donc deux listes de concepts :

- la première correspond à la liste des concepts candidats, avec leurs scores ;
- la deuxième correspond à la liste des correspondances, avec leurs scores.

Le tableau 18 représente ces deux listes pour deux syntagmes nominaux détectés par MetaMap.

Syntagme	Candidats	Correspondances
cardiac MRI	C0412692(Cardiac MRI) C0024485(MRI) C0205041(Cardiac) C0007144(Cardia)	C0412692(Cardiac MRI)
mediastinal CT	C0040405(CT) C0009778(Connecticut) C0025066(Mediastinal) C1278909(Mediastinum)	{C0025066(Mediastinal), C0040405(CT)}

Tableau 18 Exemple de concepts détectés par MetaMap pour deux syntagmes exemples

Pour un texte analysé par MetaMap, nous choisissons d'utiliser les meilleurs concepts sélectionnés par la liste de correspondance et les concepts de la liste des candidats dont le terme associé correspond exactement à celui du texte.

2.2 Méthode de Surface

Cette méthode se base sur l'hypothèse, proche de celle émise dans MetaMap, que seuls les termes présents dans UMLS et retrouvés avec seulement des variantes lexicales dans un texte médical, permettent d'identifier un concept. Cette hypothèse est restrictive car les données terminologiques d'UMLS, d'une part, ne couvrent pas toutes les formes textuelles possibles et, d'autre part, ne forment pas toutes des termes correctement construits.

Par exemple le concept *C0000768* se relie, entres autres, aux quatre termes suivants :

- *congenital Abnormality ;*
- *defects, Birth ;*
- *defects, Congenital ;*
- *congenital anomaly, unspecified.*

Sur ces quatre termes, trois représentent des couples de mots et ne peuvent se retrouver directement dans du texte.

Pour établir une association entre une chaîne de caractères et un concept, nous nous basons sur le postulat que, dans un texte médical, les concepts pertinents consistent en ceux représentés par les termes les plus longs. Les travaux de Baziz (Baziz *et al.*, 2005) utilisent notamment cette hypothèse. Par exemple, dans la séquence *'Images of right middle lobe'*, le concept pertinent à extraire doit correspondre au terme *'right middle lobe'* et non au terme *'lobe'*. Ainsi, nous évaluons la présence dans UMLS des groupes de mots de chaque phrase dans l'ordre décroissant de leur taille. Bien entendu, une instance (forme textuelle) d'un concept ne peut ni chevaucher ni contenir une instance d'un autre concept.

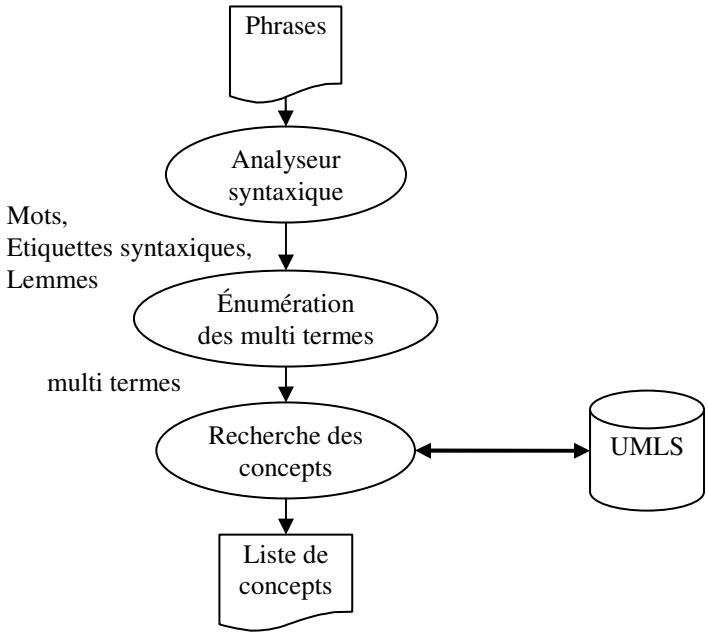


Figure 47 Extraction des concepts

Avant d'effectuer cette correspondance, ce système analyse les textes à l'aide d'un analyseur syntaxique. Cet analyseur fournit des mots segmentés, étiquetés syntaxiquement et lemmatisés (cf. figure 47).

UMLS ne contient pas toutes les formes textuelles possibles d'un concept, la correspondance stricte ne permet pas d'extraire tous les concepts. Pour dépasser cette limite, nous proposons de prendre en compte deux types de variations : la variation au niveau de la casse et la variation au niveau lexical.

Au niveau de la casse, cette méthode applique deux procédés. Le premier procédé ne respecte pas la casse, tous les mots sont transformés en minuscules. Le second utilise la casse dans le but de détecter correctement certains concepts, par exemple ceux représentant des acronymes. La méthode cherche d'abord à faire correspondre une forme textuelle d'UMLS en respectant la casse ; si aucune forme ne correspond, la méthode teste alors des correspondances avec des variations de casse (notamment des mises en majuscules).

images	of	right
Images	Of	Right
IMAGES	OF	RIGHT
image		
Image		
IMAGE		

Tableau 19 Ensemble des variations possibles pour trois mots

Au niveau lexical, cette méthode sélectionne chaque mot selon une liste de priorité : d'abord sous sa forme d'origine, ensuite sous sa forme lemmatisée. Le tableau 19 décrit le détail de ces variations.

Ne pas respecter la casse introduit du bruit au moment de l'extraction des concepts. En effet, certains mots outils deviennent équivalents à certains acronymes ou abréviations. Par exemple, l'article '*a*' correspond à la forme textuelle '*A*' du concept '*Autopsie*'. Pour résoudre ce problème,

nous émettons l'hypothèse qu'un mot-outil seul ne peut instancier un concept ; pour cela, nous nous basons sur les étiquettes syntaxiques fournies par l'analyseur morphosyntaxique.

Pour réduire le problème de l'ambiguïté sur le méta-thésaurus, nous adaptons UMLS aux besoins de notre application. En effet, certains thésaurus qui constituent UMLS sont composés d'informations trop précises, ou portent sur un domaine non pertinent par rapport à notre application. Dans ce sens, Huang (Huang *et al.*, 2003) a montré que la sélection de certaines ressources par type de rapport médical permet d'améliorer la détection des concepts. Nous émettons l'hypothèse que la suppression de certains thésaurus permet de réduire le degré d'ambiguïté au sein même du méta-thésaurus. Partant de cette hypothèse, nous éliminons les types sémantiques utilisés dans UMLS dont le domaine ne correspond pas à notre tâche. Par exemple, cette hypothèse permet de supprimer le type sémantique '*Geographic Area'* qui représente les éléments géographiques ayant une frontière. Par cette suppression, la forme '*CT*' s'associe correctement au concept '*scanner aux rayons X'*, sans s'associer au concept '*république d'Afrique centrale'* qui possède le code de pays *CT*.

Nous proposons deux variations de cette méthode, l'une utilisant l'analyseur morphosyntaxique TreeTagger[16], l'autre utilisant l'analyseur syntaxique MiniPar[17] (Lin, 1998). La premiere méthode permet d'extraire des concepts sur plusieurs langues. La deuxième fournit une analyse en dépendance complète qui sera par la suite utilisée pour calculer des scores de confiance sur les relations sémantiques.

2.3 Score de confiance

Sur les différentes méthodes d'extraction de concepts, seule l'extraction **MetaMap** fournit un score représentant la confiance dans l'extraction des concepts. Ce score se calcule à l'aide de différents paramètres :

- La centralité : favorise les concepts qui contiennent la tête du syntagme.
- La variation : évalue la variation de la chaîne textuelle par rapport à la chaîne dans UMLS.
- La couverture : nombre de mots en commun entre le syntagme et la chaîne d'UMLS.
- La cohésion : cohésion des mots participant à la chaîne de UMLS (mots continus).

MetaMap fournit un score normalisé entre 0 et 1000. Le score de confiance utilisé dans les modèles constitue la normalisation de ce score entre 0 et 1 obtenu par division du score par 1000.

2.4 Synthèse

Nous présentons trois méthodes pour instancier la fonction $Ind_{concept}^{ph}$. Ces méthodes permettent de détecter les concepts à partir du texte médical. Dans la suite, nous référons à ces différentes méthodes à l'aide des noms suivants :

- **MetaMap** : la détection de concepts à l'aide de MetaMap ;
- **MapTreeTagger** : la détection à l'aide de la méthode de surface utilisée avec TreeTagger ;
- **MapMiniPar** : la détection à l'aide de la méthode de surface utilisée avec MiniPar.

Sur ces méthodes, seule la méthode MetaMap fournit un score de confiance sur le processus d'extraction. Dans les autres méthodes, ce score n'est pas utilisé pour concepts.

---

[16] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

[17] http://www.cs.ualberta.ca/~lindek/minipar.htm

3 Détection des relations

Suite à la détection des concepts, nous construisons la représentation de phrase par ajout de relations sémantiques. Les relations sémantiques utilisées sont celles définies par le type relation, elles correspondent à celles du réseau sémantique d'UMLS. La fonction de détection des relations se base sur l'hypothèse qu'une relation existe entre deux concepts d'un document si ces deux concepts apparaissent dans la même phrase et si le réseau sémantique d'UMLS définit cette relation sémantique. Dans le réseau sémantique, les relations relient des classes de concepts. Pour détecter les relations entre concepts, nous étiquetons chaque concept par son type sémantique dans UMLS. Le système détecte une relation sémantique entre deux concepts si le réseau sémantique définit une relation entre les types sémantiques de ces deux concepts.

Par exemple : Dans la phrase *'Show me chest CT images with emphysema'* le système détecte le concept *C0202823(chest CT)* qui se relie au type sémantique *T060(Diagnostic Procedure).* On détecte ensuite le second concept *C0034067(emphysema)* relié au type sémantique *T047(Disease or Syndrome)*. Dans le réseau sémantique, Les types sémantiques T060 et T047 peuvent se relier par les relations sémantiques :

T151(affects), T162(measures), T163(diagnoses), T166(associated_with)

On détecte alors les quatre relations suivantes entre les deux concepts de la phrase :

(C0202823, T151, C0034067), (C0202823, T162, C0034067), (C0202823, T163, C0034067), (C0202823, T166, C0034067)

3.1 Score de confiance

Les relations que nous détectons sont plus ou moins ambigües et une incertitude existe sur leur existence dans le texte. Entre deux concepts, si plusieurs relations peuvent apparaître, probablement qu'une partie seulement de ces relations apparaît effectivement dans la phrase. Nous proposons d'utiliser des indices syntaxiques pour calculer le score de confiance dans une relation étiquetée. Pour cela nous utilisons les chemins syntaxiques qui relient les deux concepts de la relation et nous proposons de modéliser ces chemins pour calculer le score de confiance.

3.1.1 Expression syntaxique des relations sémantiques

Des travaux proposent l'utilisation d'information syntaxique pour extraire des relations (Lin, 1998) (Delbecque *et al.*, 2005). Dans (Delbecque *et al.*, 2005) les auteurs extraient les relations entre les concepts d'une même phrase et utilisent des informations provenant des verbes de la phrase pour déterminer ou non l'existence de la relation sémantique.

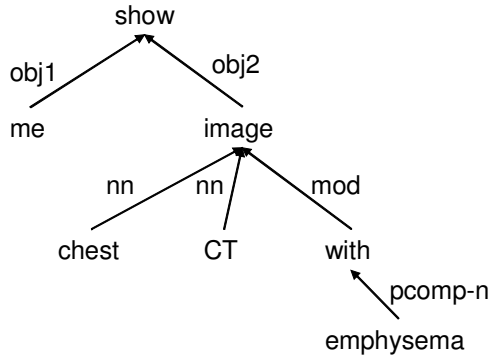
Dans notre méthode, lors de la détection des concepts, chaque concept est associé à une séquence de lemmes, par exemple un concept est associé à la séquence *{chest, CT, emphysema}*. Nous proposons de détecter une relation sémantique entre deux concepts en fonction de la relation syntaxique fournie par un analyseur syntaxique en dépendance qui relie ces deux concepts. Nous nommons cette relation un *chemin syntaxique*.

Un chemin syntaxique constitue l'ensemble des lemmes et des relations syntaxiques qui forment le chemin reliant deux lemmes dans l'arbre de dépendance d'une phrase. Les concepts correspondent à un ensemble de lemmes ; le chemin syntaxique entre deux concepts forme le chemin dans l'arbre de dépendance qui relie les têtes syntaxiques des deux concepts.

Les concepts tels que nous les détectons à l'aide de nos méthodes ne représentent pas obligatoirement des sous-arbres corrects de l'analyse en dépendance. Pour sélectionner la tête d'un

groupe de lemmes correspondant à un concept, nous prenons simplement le nœud de l'ensemble le plus proche de la racine. Par exemple, si un concept se relie à l'ensemble de termes *{chest, CT, image}* dans un arbre de dépendance tel que décrit dans la figure 48, alors la tête du concept correspond simplement au lemme *'image'*. Si deux nœuds se positionnent à égale distance de la racine alors nous sélectionnons le nœud le plus à droite dans la phrase comme représentant la tête du concept.

Par exemple soit le concept c_1 associé à *{chest, CT}* et le concept c_2 associé à *{emphysema},* dans la figure 48, le chemin syntaxique entre ces deux concepts c_1 et c_2 est le chemin qui relie *'CT'* à *'emphysema'* dans l'arbre de dépendance. Ce chemin est donc *(nn)image(mod)With(pcomp-n)* sachant l'analyse de la phrase présentée dans la figure 48. Si deux concepts s'associent à la même tête alors nous utilisons la pseudo-relation *'ident'* entre ces deux concepts.



relation	Chemin syntaxique
(C0817096, T135, C0034067) (chest, location of , emphysema)	(nn)image(mod)with (pcomp-n)
(C0202823, T135, C0817096) (chest, location of, chest CT)	(ident)

Figure 48 Deux exemples de chemins syntaxiques extraits sur un arbre de dépendance produit par MiniPar

3.1.2 Calcul de la confiance par des modèles de chemin syntaxique

Nous émettons l'hypothèse que la probabilité d'un type de relation sémantique *Rs* pour une relation *r* telle que *(type(r)=Rs)* entre deux concepts associés aux têtes s_0 et s_{n+1} peut s'estimer par le chemin syntaxique $ch=s_1,\ldots,s_n$ entre ces deux têtes. Nous construisons un modèle de chemin pour chaque type de relation sémantique définie. Nous calculons la probabilité pour une relation *r* avec le type *Rs* que le chemin syntaxique *ch* qui lui correspond soit généré par le modèle de ce type de relation M_{Rs}.

$$P(type(r)=Rs|p)\propto P(p|M_{Rs})$$

où M_{Rs} dénote le modèle des relations étiquetées par le type *Rs*

Nous considérons qu'un chemin syntaxique s'associe à un type de relation sémantique *Rs* si sa probabilité d'être généré par le modèle M_{Rs} du type de relation *Rs* donne un résultat supérieur à sa probabilité d'être généré par un modèle générique de relation sémantique M_{Rg}.

$$P(ch,M_{Rs})>P(ch,M_{Rg})$$

Nous proposons deux modèles pour calculer cette probabilité :

- Modèle **unigramme** : un chemin syntaxique se représente comme un ensemble d'éléments indépendants (modèle unigramme) $ch = \{elem\}$, où les *elem* désignent indifféremment des lemmes ou des relations. La probabilité d'un chemin *ch* pour un type de relation sémantique *Rs* s'écrit alors :

$$P(ch|M_{Rs}) = \prod_{elem \in ch} P(elem|M_{Rs})$$

Nous estimons ensuite la probabilité d'un élément de syntaxe *elem* sachant le modèle de relation M_{Rs} en calculant le rapport entre *N(elem,Rs)* le nombre d'occurrences de *elem* dans une relation sémantique *Rs* tel que *type(r)=Rs* du corpus d'apprentissage et $N(*,Rs)$ la fréquence de tous les éléments formant l'ensemble des relations de type *Rs* soit :

$$P(ch|M_{Rs}) = \prod_{elem \in ch} \frac{N(elem, Rs)}{N(*, Rs)}$$

- Modèle **bigramme** : nous construisons un modèle où le chemin se forme comme un ensemble de couples $ch' = \{(lem, Rsyn)\}$ constitués de *lemmes* et de relations syntaxiques *Rsyn*. La probabilité d'un chemin *ch'* pour un type relation *Rs* s'écrit alors :

$$P(ch'|M_{Rs}) = \prod_{(lemme,Rsyn) \in ch'} (\lambda) P((lem, Rsyn)|M_{Rs}) + (1-\lambda) P(lem|M_{Rs}) P(Rsyn|M_{Rs})$$

Nous estimons ensuite la probabilité d'un élément de syntaxe *lem* ou *Rsyn* comme dans le modèle unigramme. La probabilité d'un couple sachant le modèle de relation M_{Rs} s'estime par le rapport entre *N((lem,Rsyn),Rs)* et le nombre d'occurrences du couple *(lem,Rsyn)* dans une relation sémantique *r,* telle que type(r)=*Rs,* du corpus d'apprentissage et $N((*,*),Rs)$ la fréquence de tous les couples formant l'ensemble des relations de type *Rs*, soit :

$$P((lemme, Rsyn)|M_{Rs}) = \frac{N((lemme, Rsyn), Rs)}{N((*,*), Rs)}$$

Au regard de ces modèles, nous considérons la validité d'un chemin syntaxique pour un type de relation *Rs* si la probabilité de générer ce chemin à l'aide du modèle de cette relation M_{Rs} est supérieure à la probabilité de générer ce chemin à l'aide d'un modèle de relations génériques. Nous calculons donc le score de vraisemblance suivant pour établir la confiance d'une relation :

$$conf(ch, Rs) = \frac{P(ch, M_{Rs})}{P(ch, M_{Rg})}$$

où $P(ch, M_{Rg})$ dénote la probabilité du chemin syntaxique *ch* pour le modèle de relations génériques. Ce modèle s'obtient sur l'ensemble des relations sémantiques du corpus.

3.2 Synthèse

Nous présentons une méthode qui permet d'extraire les relations sémantiques à partir d'un ensemble de concepts $Ind^{ph}_{relation}(ph, DV^{ph}_{concepts})$. L'ensemble de concepts utilisés dans cette étape de détection de relations provient indifféremment de l'une de nos méthodes de détection des concepts.

Nous proposons une méthode pour calculer un score de confiance dont nous donnons deux variations. Cette méthode repose sur une analyse syntaxique en dépendance, or seule l'utilisation de la

méthode MapMiniPar fournit cette analyse lors de l'extraction des concepts. Nous ne pourrons tester l'utilisation de ce score de confiance qu'avec une détection des concepts par MapMiniPar.

4 Conclusion

Cette partie développe une application des deux modèles proposés aux textes médicaux. Elle définit le support de types de ces modèles à l'aide de la source de connaissances UMLS. Cette partie fournit aussi une implémentation du processus d'indexation intermédiaire.

Cette application des deux modèles se base sur quelques simplifications :

- Elle ne prend pas en compte les référents, par conséquent un type de concept n'apparaît qu'une seule fois au sein du graphe final du document. Chacune des détections d'un concept dans une représentation de phrase correspond à celles des autres représentations de phrases. La fonction de correspondance se simplifie en conséquence du fait qu'elle n'évalue plus l'égalité des référents lors de la correspondance des concepts.
- La non utilisation du treillis a pour conséquence de simplifier la correspondance du modèle local *ML*, celle-ci passe d'une projection du graphe conceptuel de la requête à une simple intersection de graphes.

Au final, l'utilisation d'UMLS nous permet d'établir une instance du modèle pour le domaine médical. Cette ressource montre un potentiel certain car elle recouvre un domaine vaste, elle permet l'indexation automatique des différentes représentations. Nous proposons des méthodes pour construire les représentations intermédiaires des documents sur le corpus. Ces représentations sont importantes car elles sont à la base de la construction de nos modèles, toutes les erreurs dans ces représentations sont conservées dans les représentations suivantes.

Par la suite, nous évaluons cette application des modèles. Pour cela nous choisissons une collection d'évaluation sur laquelle nous évaluons les deux modèles. Sur cette collection, nous testons au préalable les différentes fonctions de détection des concepts et des relations permettant la création de la représentation intermédiaire, cela afin de les comparer et de contrôler leur efficacité. Ensuite nous testons les deux modèles proposés.

Bilan

Cette partie décrit une méthode d'indexation permettant d'appliquer les modèles proposés sur des textes médicaux. Cette méthode décompose l'application des modèles en deux parties, la première consiste à créer une représentation pour chaque phrase du document, cette étape est dépendante du domaine sur lequel s'appliquent les modèles. À partir de cette représentation du document nous fournissons des méthodes génériques qui permettent de créer des représentations des documents conformes aux modèles. Cette approche a l'avantage de mutualiser les processus utilisés sur le texte. Par conséquent, adapter les modèles à un nouveau domaine ne nécessite que la définition des types concepts $T_{concepts}$, relations $T_{relations}$ et référents $T_{référents}$ sur ce domaine et la construction des méthodes permettant la production de la représentation intermédiaire du document.

Le premier chapitre de cette partie définit la représentation intermédiaire des documents utilisés pour le texte. Cette représentation se base sur un support de vocabulaires proche de celui utilisé par le modèle local et elle met en avant l'utilisation d'un score de confiance. À partir de cette représentation, nous proposons des méthodes permettant de créer les représentations finales des documents et des requêtes pour les deux modèles.

- Le modèle local effectue la concaténation des représentations de phrase du document. Il utilise des mesures habituellement utilisées dans le modèle vectoriel. Il répercute aussi sur la représentation finale les scores de confiance de chaque phrase.
- Le modèle global utilise des estimations permettant la création d'un modèle de graphe à partir des représentations de chaque phrase du document.

Les modèles utilisés s'obtiennent donc par des méthodes différentes qui considèrent des visions divergentes de l'obtention du thème d'un document. Nous fournissons dans l'annexe C la représentation d'un document par le modèle intermédiaire, le modèle local et le modèle global, cela à l'aide d'un formalisme XML.

Le deuxième chapitre de cette partie détaille les méthodes qui permettent de créer la représentation intermédiaire des documents sur le domaine médical. Cette représentation se base sur un support de types défini en utilisant la source de connaissances UMLS. Cette source de connaissances définit des concepts et des relations sémantiques entre concepts, que nous utilisons pour créer le support de type. Cette source de connaissances fournit un certain nombre d'informations permettant la détection des concepts à partir du texte, notamment en fournissant un certain nombre de termes associés à ces concepts. Nous proposons trois méthodes pour détecter les représentations des phrases. Ces méthodes se basent sur trois approches différentes pour détecter les concepts d'une phrase, et sur une méthode commune pour détecter les relations à partir de ces concepts. Sur ces méthodes de détection de concepts, l'une se base sur un outil disponible pour la détection de concepts, les deux autres se basent sur une analyse morphosyntaxique de la phrase et sur une correspondance de termes. Nous utilisons pour cette méthode deux variations qui utilisent chacune un analyseur morphosyntaxique différent (TreeTagger et MiniPar).

Nous calculons un score de confiance pour les concepts uniquement avec la méthode MetaMap, seule méthode fournissant un score de confiance lors de la détection des concepts. Nous calculons

aussi un score de confiance sur les relations sémantiques pour la méthode qui utilise l'analyseur syntaxique MiniPar.

Par la suite nous proposons des expérimentations qui évaluent les performances des deux modèles présentés lors de leur application au domaine médical. Ces évaluations testent les différentes parties de la construction des modèles et comparent leurs performances, notamment en termes de précision.

PARTIE 5 : EXPERIMENTATIONS DES MODELES

Introduction

Nous présentons dans cette thèse deux modèles d'expressivité forte. Ces deux modèles constituent des modèles orientés précision dont le but est de résoudre des besoins experts. Pour chacun de ces modèles nous détaillons les caractéristiques de leur application sur du texte et plus précisément sur du texte médical. Dans cette partie, nous évaluons l'intérêt de ces modèles. En recherche d'information, l'évaluation des systèmes passe par l'utilisation de collections de tests. Ces collections sont formées d'un corpus de documents (ici nous utilisons un corpus de documents médicaux) et d'un jeu de requêtes résolues, c'est-à-dire des requêtes associées à une liste de documents pertinents sur le corpus. Cette démarche est importante car elle permet de confronter les systèmes proposés à la réalité mais aussi à d'autres systèmes. Nous utilisons ici la collection CLEF médicale sur laquelle nous effectuons un certain nombre d'évaluations.

Ce chapitre présente d'abord la collection CLEF médicale, puis présente les expérimentations mises en place sur les différents modèles.

1 La collection CLEF médicale

Nous proposons une instanciation des modèles au domaine médical, nous expérimentons donc ces modèles sur une tâche de recherche d'information médicale. Nous utilisons pour cela les données de la campagne d'évaluation CLEF médicale à laquelle nous avons participé en 2007. Ces données définissent le corpus et les requêtes sur lesquelles se basent les expérimentations. Ces données permettent de tester nos modèles et nos méthodes dans un cadre réel. Participer à des campagnes d'évaluation internationale permet de plus de positionner nos systèmes par rapport aux systèmes proposés par d'autres universités. Ce cadre nécessite également le développement de méthodes efficaces capables de produire des résultats dans les délais impartis par la campagne d'évaluation.

1.1 Présentation

La tâche ImageCLEFmed (clef médicale) constitue une tâche de la campagne d'évaluation CLEF (Cross Language Evaluation Forum). Cette campagne d'évaluation a pour but d'évaluer l'accès à des données multilingues. La tâche CLEF médicale appartient depuis 2004 à la campagne d'évaluation CLEF. Le but principal de cette tâche consiste en l'amélioration de la recherche d'images médicales sur des documents hétérogènes et multilingues contenant des images et du texte.

La collection CLEF médicale (Müller *et al.*, 2007) se compose d'images et de diagnostics associés à ces images. Ces diagnostics utilisent trois langues différentes : le français, l'anglais et l'allemand. Cette collection est formée par le regroupement de plusieurs sous-collections :

- **Casimage** qui contient 9 000 images associées à 2 000 cas avec un diagnostic par cas. Sur cette collection, la majorité des diagnostics utilisent le français, les 20 % restant l'anglais.
- **PEIR** (Pathology Education Instructional Resource), 33 000 images associées à des annotations en anglais provenant du projet HEAL (Health Education Assets Library) et qui sont constituées essentiellement de pathologies.

- **MIR** (Mallinkrodt Institute of Radiology) qui contient environ 2 000 images de médecine nucléaire associées à des annotations en anglais.
- **PathoPic** (Pathology images) formée d'environ 9 000 images avec de longues annotations en allemand et la traduction en anglais.
- **MyPACS** formée de 3 577 cas en anglais associés à 15 140 images.
- **CORI** (Clinical Outcomes Research Initiative) un ensemble de 1 496 images d'endoscopie annotées en anglais.

Ainsi réunies, ces sous-collections forment un ensemble de 66 662 images associées à 55 485 annotations, le tableau 20 résume ces informations.

Collection	Cas	Images	Annotations	Annotations par langue
Casimage	2076	8725	2076	Français - 1899 Anglais - 177
MIR	407	1177	407	Anglais - 407
PEIR	32319	32319	32319	Anglais - 32319
PathoPic	7805	7805	15610	Allemand – 7805 Anglais - 7805
MyPACS	3577	15140	3577	Anglais - 3577
CORI	1496	1496	1496	Anglais - 1496
Total	47680	66662	55485	

Tableau 20 Détails des collections de CLEF image médicale

Nous remarquons que certaines collections se focalisent sur les *cas* et que d'autres se focalisent sur les *images*. Cependant, comme dans la collection PEIR, certaines images des collections *image* utilisent plusieurs fois le même diagnostic comme annotation.

Sur ces collections, la campagne CLEF médicale fournit chaque année un ensemble de requêtes. Ces ensembles de requêtes n'utilisent pas toutes les collections. Le tableau 21 résume les ensembles de requêtes et les collections utilisées par ces requêtes entre 2005 et 2007.

Année	2005	2006	2007
Collections	Casimage, MIR, PEIR, PathoPic		Casimage, MIR, PEIR, PathoPic, MyPACS, CORI
Nombre de requêtes	25	30	30

Tableau 21 Détails des données d'évaluation sur les différentes années

Une requête du corpus CLEF médicale est constituée d'une partie textuelle exprimée dans les trois langues du corpus (anglais, français, allemand), accompagnée d'images exemples. Ces requêtes sont pertinentes pour notre application car elles expriment des besoins experts. Dans nos expérimentations, nous n'utilisons que la partie textuelle des requêtes. Nous montrons un exemple de requête dans la figure 49. Les campagnes de 2006 et 2007 établissent le classement des requêtes selon 3 aspects :

- **Visuel :** les requêtes qui portent essentiellement sur des aspects visuels.
- **Mixte :** les requêtes qui portent sur des aspects visuels et textuels.
- **Textuel :** les requêtes qui portent essentiellement sur des aspects textuels.

Show me x-ray images with fractures of the femur.
Zeige mir Röntgenbilder mit Brüchen des Oberschenkelknochens.
Montre-moi des fractures du fémur.

Figure 49 Exemple de requête de CLEF Médicale

1.2 Évaluation sur corpus

Pour des questions de facilité, certains de nos processus ne sont évalués que sur des parties ou sur des versions allégées de la collection CLEF médicale. D'une part, certaines expérimentations ne portent que sur la partie anglaise du corpus, nous évaluons ces expérimentations seulement sur les éléments pertinents de la collection appartenant à la langue anglaise. D'autre part, certains résultats de recherche d'information sont évalués directement au niveau diagnostic, les méthodes de création de graphes ne prenant pas en compte les images. Enfin, dans des expérimentations sur les diagnostics, nous éliminons les diagnostics redondants.

Le corpus de CLEF médicale ne fournissant que des évaluations de pertinence au niveau des images, par la suite, nous considérons qu'un diagnostic est pertinent si le corpus désigne au moins une des images qui lui correspond comme pertinente. Les résultats de nos méthodes peuvent alors s'évaluer directement au niveau textuel.

Les différentes évaluations menées sur le corpus n'utilisent pas systématiquement les mêmes années. Dans certains cas, nous découpons l'ensemble des requêtes en un corpus d'apprentissage et un corpus de tests. Pour chacune de nos expérimentations nous détaillons donc les années utilisées.

Pour décrire les résultats d'une expérimentation, nous nous référons à ces évaluations à l'aide de noms composés de trois éléments, le premier définit la langue :

- **EN** pour l'utilisation de l'anglais
- **A** pour l'utilisation de toutes les langues

Le deuxième définit le niveau de l'évaluation :

- **IMG** pour l'évaluation au niveau des images.
- **TXT** pour l'évaluation au niveau du texte.
- **DIAG** pour l'évaluation avec filtrage des diagnostics redondants.

Enfin, le dernier élément du nom de l'évaluation fournit l'ensemble des requêtes utilisées par années.

Par exemple : une évaluation des résultats au niveau du texte sur la partie anglaise de la collection, avec les requêtes de 2005 et de 2006 s'écrira :

EN_TXT_0506

1.3 Synthèse

La collection CLEF médicale fournit le corpus et les requêtes sur lesquelles nous utilisons les deux modèles de recherche d'information proposés. En fonction des éléments de la collection choisie, le contexte d'expérimentation de ces modèles se définit par :

C={ensemble de textes ou images de CLEF image}

q=requête de CLEF image

Par exemple : pour une expérimentation sur la partie anglaise du corpus évaluée avec les requêtes de CLEF médicale 2005 et 2006, le contexte se note :

C={ensemble de textes de Casimage, MIR, PEIR, PathoPic }

q=requêtes de CLEF 2005 ou 2006

Un premier intérêt de cette collection est qu'elle permet de tester l'utilité de la structure conceptuelle sur une tâche de recherche d'information médicale. Le second intérêt provient de son caractère multilingue qui permet d'évaluer l'aspect multilingue des structures conceptuelles que nous utilisons.

2 Évaluations

Les éléments à évaluer dans les modèles proposés sont multiples. Nous présentons d'abord les outils utilisés pour nos expérimentations puis nous définissons le protocole régissant l'enchaînement de nos expérimentations.

2.1 Outils d'expérimentation

Pour effectuer les expérimentations, nous utilisons des procédés qui décomposent l'expérimentation en un ensemble de processus indépendants. Une expérimentation est donc composée par l'enchaînement de différents processus. Cette méthode s'inspire et utilise les méthodes proposées dans (Chevallet, 2004). La flexibilité de cette méthode d'expérimentation permet de faire varier de nombreux paramètres expérimentaux. Même si cette méthode n'optimise pas le temps des traitements, elle à l'avantage de permettre d'expérimenter seulement des parties du traitement sans avoir à effectuer toute la chaîne de traitement et de permettre la réutilisation des programmes. Par exemple, dans notre méthode, seulement la première partie du traitement dépend du domaine. L'indépendance des traitements nous permet d'adapter nos expérimentations à un autre domaine en ne programmant ou en n'utilisant que des outils permettant la création de la représentation intermédiaire, le reste du traitement restant le même.

Nous évaluons nos expérimentations à l'aide du programme *trec_eval*[18], programme d'évaluation proposé par les campagnes d'évaluation de recherche d'information. Dans ces expérimentations, nous évaluons les résultats à l'aide de la MAP (précision moyenne) du fait que cette pondération donne une appréciation globale des performances du système, et à l'aide de la précision à 5 documents (P@5) du fait que cette mesure témoigne des performances du système sur les premiers documents retrouvés.

[18] http://trec.nist.gov/data/reljudge_eng.html

Ces derniers correspondent aux documents les plus regardés par l'utilisateur en recherche d'information orientée précision.

2.2 Protocole

2.2.1 Présentation

Les différentes expérimentations que nous menons sur la collection CLEF ont un double objectif. Le premier objectif est d'évaluer les propositions de la thèse et le second de sélectionner un système de recherche d'information efficace qui soit utilisable lors de la campagne d'évaluation CLEF médicale. Nous présentons les expérimentations en trois étapes telles que décrites à travers le tableau 22. Pour chacune de ces expérimentations nous conservons les trois méthodes de détection des concepts, ces méthodes possédant chacune leurs spécificités comme nous le détaillons dans le premier chapitre de ces expérimentations. La première étape permet de valider la création des représentations intermédiaires, la deuxième évalue les différentes variations du modèle local, et enfin la dernière évalue les variations du modèle global.

Étapes	Évaluation	MapTreeTagger	MapMiniPar	MetaMap
Représentation intermédiaire	Détection des concepts : modèle conceptuel	X	X	X
	Détection des relations : modèle relationnel	X	X	X
Modèle local	Sans score de confiance	X	X	X
	Avec un score de confiance sur les concepts			X
	Avec un score de confiance sur les relations		X	
Modèle global	Sans étiquettes		X	
	Avec étiquettes	X	X	X
	Avec étiquettes, avec plusieurs détections	X (CLEF 2007)		

Tableau 22 liste des expérimentations

2.2.2 Représentation intermédiaire

La construction des modèles se base sur une première étape qui consiste à extraire des représentations de phrase pour chaque phrase des documents et des requêtes. Sur cette étape nous proposons plusieurs méthodes d'extraction des représentations de phrase. Avant d'évaluer les deux modèles proprement dits, nous évaluons de manière séparée les performances de la génération des représentations de phrase. Pour cela nous testons des modèles à base de concepts et des modèles relationnels de niveau sémantique. Nous extrayons ces éléments sur les phrases en utilisant nos trois méthodes d'extraction des concepts qui fournissent des surreprésentations des documents. Sur ces expérimentations, nous nous interrogeons sur l'intérêt des informations de niveau sémantique par rapport aux informations de niveau plus lexical. Nous comparons ainsi les performances de ces modèles par rapport à des modèles à base de lemmes. Nous évaluons les caractéristiques et les variations des différentes méthodes d'extraction des concepts et leur impact sur la détection des relations. Nous comparons ensuite les résultats des concepts avec ceux des relations.

2.2.3 Modèle local

Une fois la représentation intermédiaire validée, nous évaluons les caractéristiques du modèle local. Sur ce modèle nous conservons les trois méthodes de détection des concepts et pour chacune de ces méthodes nous testons plusieurs variations de pondérations, que se soit pour les relations ou pour les concepts. Les méthodes de détection des concepts ne permettant pas toutes d'obtenir un score de confiance, nous testons le modèle local d'une part sans score de confiance et nous le testons d'autre part avec la prise en compte de ce score quand cela est possible (cf. tableau 22). Nous testons

l'utilisation de scores de confiance sur les concepts détectés par la méthode MetaMap et sur les relations sémantiques extraites par la méthode MapMiniPar. Ces expérimentations nous permettent de déterminer l'intérêt de ce score en recherche d'information. Au final nous donnons une synthèse des résultats pour ce modèle.

2.2.4 Modèle global

Nous testons ensuite le modèle global et nous évaluons les différentes variations du modèle de graphe utilisé pour représenter les documents. Pour commencer nous testons les performances de ce modèle sans prise en compte des étiquettes. Nous comparons ces premiers résultats à un autre modèle applicable sur les graphes et à un modèle de langue conceptuel. Nous évaluons ensuite le modèle complet avec prise en compte des étiquettes. Nous effectuons ensuite quelques expérimentations supplémentaires pour déterminer les types de requêtes les plus adaptés à l'utilisation du modèle global. Enfin, ce modèle en fournissant la possibilité, nous testons l'intérêt du regroupement des différentes extractions sur une expérimentation multilingue. Cette expérimentation fournit de bonnes performances et permet le regroupement des requêtes, nous l'avons ainsi sélectionnée pour la campagne d'évaluation CLEF où elle nous a permis d'obtenir les meilleurs résultats parmi 13 équipes.

2.2.5 Synthèse

Nous finissons la partie expérimentations par la comparaison des deux approches. Nous réalisons ensuite un bilan au niveau des modèles dans lequel nous fournissons des pistes de réflexion sur l'impact de l'utilisation du support. Nous établissons également un bilan sur la campagne d'évaluation.

Chapitre X Représentation Intermédiaire

Les parties précédentes développent deux modèles de recherche d'information dont l'obtention passe par la détection d'une représentation intermédiaire des documents. Cette représentation intermédiaire constitue la base du processus d'indexation de ces modèles. Son obtention est une étape importante pour laquelle nous proposons plusieurs méthodes. Cette partie évalue les différents aspects de ces méthodes de génération sur des textes médicaux (présentés dans le chapitre IX), d'une part la génération des concepts et d'autre part celle des relations.

1 Mise en œuvre

La production et l'évaluation de la représentation intermédiaire ont nécessité l'implémentation des méthodes de détection des concepts et de détection des relations. Nous avons créé la méthode de détection des concepts afin de l'appliquer sur des données de différents analyseurs morphosyntaxiques, ici TreeTagger et MiniPar.

Au niveau du temps d'exécution, la méthode de surface se décompose en deux parties : l'analyse morphosyntaxique puis la détection des concepts. Le temps d'exécution de l'analyse morphosyntaxique varie en fonction de l'analyseur. TreeTagger fournit une analyse rapide de la collection alors que MiniPar, qui extrait un arbre de dépendance complet, est beaucoup plus long (environ six heures sur la collection[19]). La détection des concepts quant à elle fournit une analyse de la collection en moins de deux heures.

Sur la méthode de surface aussi bien que sur les méthodes utilisant l'outil MetaMap, nous avons effectué un certain nombre d'expérimentations pour déterminer les paramètres adéquats, et les informations utiles à la sélection des concepts. Cela représente environ trois mois pleins de programmation et d'expérimentation.

2 Détection des concepts

L'objectif des expérimentations présentées dans cette section consiste à évaluer les fonctions de détection de concepts. Dans ce but, nous effectuons une comparaison des résultats entre les différentes méthodes d'extraction des concepts avec les résultats d'un modèle de référence basé sur les lemmes. Les expérimentations de cette partie n'utilisent que la partie anglaise du corpus et les requêtes de CLEF 2005 en anglais. Nous évaluons les résultats au niveau image A_IMG_05 (collection 2005, toutes les langues et évaluation au niveau des images).

[19] sur un P4 3.2 GHz avec une RAM de 2Go

Sur cette partie du corpus nous testons nos trois méthodes d'extraction des concepts. Les deux premières utilisent la méthode de surface appliquée pour l'une avec TreeTagger (MapTreeTagger) et pour l'autre avec MiniPar (MapMiniPar). Sur ces deux méthodes nous utilisons les filtrages présentés dans la section 2.2 du chapitre IX, que nous nommons :

- F1 : le filtrage effectué sur les étiquettes syntaxiques;
- F2 : le filtrage effectué sur les types sémantiques;
- F3 : le filtrage de certains thésaurus[20].

Nous présentons ensuite la méthode qui utilise l'outil MetaMap.

2.1 Méthode

Nous évaluons les performances des méthodes de détection de concepts à l'aide d'un modèle d'indexation conceptuel *IC*. Ce modèle se base sur un vocabulaire de concepts. Les unités de ce vocabulaire qui forment la représentation du document sont celles que le système détecte à l'aide de l'une de nos méthodes de détection des concepts. Ce modèle utilise un vocabulaire à pondération unique. Deux méthodes pour le calcul de cette pondération peuvent s'appliquer sur les concepts : la première consiste en une variation du *tf.idf*, la seconde correspond à la *divergence from randomness* (*DFR*)[21] (Amati et van Rijsbergen, 2002). Le modèle se définit comme :

$$IC = \left(ST_{UMLS}, SVD_{IC}, SVQ_{IC}, RC_{IC}\right)$$

où $SVD_{IC} = SVQ_{IC} = \left(V^{IC}_{concepts}\right)$ avec $V^{IC}_{concepts} \subseteq V_{concepts} \times P^1$

$DV^{IC}_{concepts} \subset V^{IC}_{concepts}$ constitue l'ensemble des concepts détectés dans les phrases du document et P^1 *le tf.idf* ou la *DFR*

$QV^{IC}_{concepts} \subset V^{IC}_{concepts}$ constitue l'ensemble des concepts détectés dans la requête et P^1 est leur fréquence.

RC_{IC} effectue le produit scalaire.

2.2 Méthode de référence

Nous comparons les résultats des méthodes de détection des concepts à une méthode de référence. Cette méthode (Chevallet *et al.*, 2005) se base sur un modèle à base de lemmes *IL* obtenu avec l'analyseur TreeTagger après un filtrage des types syntaxiques. Ce filtrage ne conserve que les noms, les adjectifs et les abréviations. Ce modèle utilise les deux pondérations *tf.idf* et *DFR*. Il se représente alors par :

[20] NCI et PDQ : Ce filtrage se justifie ici par le fait que ces thésaurus portent sur des points précis de cancérologie alors que la collection considérée est plus générale, et concerne l'ensemble des pathologies.

[21] Pondération du modèle probabiliste, qui se fonde sur l'hypothèse que plus la fréquence d'un terme dans un document diverge de sa fréquence dans la collection plus ce terme est important.

$$IL = \left(ST_{IL}, SVD_{IL}, SVQ_{IL}, RC_{IL}\right)$$

où $ST_{IL}=T_{lemmes}$ et $SVD_{IL} = SVQ_{IL} = \left(V_{lemmes}^{IL}\right)$ avec $V_{lemmes}^{IL} \subseteq T_{lemmes} \times P^1$

$DV_{lemmes}^{IL} \subset V_{lemmes}^{IL}$ constitue l'ensemble des lemmes du document et P^1 le *tf.idf* ou le *DFR*

$QV_{lemmes}^{IL} \subset V_{lemmes}^{IL}$ constitue l'ensemble des lemmes de la requête et P^1 leur fréquence.

RC_{IL} effectue le produit scalaire.

En utilisant tous les lemmes[22], les pondérations *tf.idf* et *DFR* fournissent respectivement une précision moyenne (MAP) de 0,1543 et de 0,1797. Sur les trois langues de la collection, cette méthode fournit une précision moyenne de 0,1725 avec le *tf.idf*. En indexant seulement la partie anglaise de la collection, les résultats diminuent de 4%. Par conséquent la partie anglaise de la collection peut s'utiliser seule dans des expérimentations.

2.3 Méthode MapTreeTagger

Lors des expérimentations, nous remarquons que le concept *'image'* dans les requêtes a un impact négatif sur la pondération *DFR*. En effet, le concept correspondant au terme *'image'* ne discrimine pas les documents dans la tâche de recherche d'image, car implicitement chaque diagnostic correspond à des images. Par conséquent, pour permettre une meilleure comparaison des résultats, nous supprimons le concept correspondant au terme *'image'*. Le tableau 23 présente d'abord les résultats de la détection des concepts les plus longs, cela à l'aide des différentes possibilités d'utilisation de la casse et de filtrages.

L'utilisation des concepts correspondant aux instances textuelles les plus longues, donne des résultats inférieurs à ceux obtenus à l'aide des mots. Cette baisse de performance s'explique par l'extrême précision des concepts détectés. En effet, des concepts comme ceux correspondant à la forme *'Right middle lobe'* ou à *'Chest CT'* semblent trop précis et n'apparaissent que dans très peu de documents. Par conséquent, leur utilisation à la place de leurs constituants entraîne une trop forte baisse du rappel.

Pour résoudre ces problèmes, nous relâchons l'hypothèse qui privilégie les formes textuelles les plus longues. Ainsi, à partir de chaque phrase, la méthode détecte les concepts correspondant à toutes les formes textuelles. L'utilisation de cette méthode (présentée dans la deuxième partie du tableau 23) améliore de 50% les résultats et permet d'atteindre et de surpasser les résultats des lemmes. Cela découle d'une augmentation du nombre de documents retrouvés dû à l'extraction de concepts plus généraux qui permettent une meilleure couverture du contenu de la requête.

[22] Excepté le mot image qui est plus proche des méta-données que du thème

	sans casse	utilisation de la casse	F1	F2	F3	Tf.idf	DFR
Les plus longues formes textuelles						0,0827	0,0945
	X					0,1001	0,0939
	X		X			0,1026	0,118
	X		X	X		0,1013	0,1161
	X		X	X	X	**0,1085**	**0,1169**
		X				0,0952	0,0908
		X	X			0,0987	0,1147
		X	X	X		0,0976	0,1149
		X	X	X	X	0,1074	0,1160
Toutes les formes textuelles						0,1185	0,1500
	X					0,1560	0,1708
	X		X			**0,1566**	**0,1859**
	X		X	X		0,1552	0,1835
	X		X	X	X	0,1541	0,1827
		X				0,1452	0,1647
		X	X			0,1473	0,1817
		X	X	X		0,1467	0,1804
		X	X	X	X	0,1469	0,1819

Tableau 23 Résultats en précision moyenne obtenus sur les concepts à l'aide de l'extraction MapTreeTagger sur A_IMG_05 sur les pondérations tf.idf et DFR

Les résultats obtenus en utilisant la casse pour établir une liste de priorités sur les formes en majuscule des termes, restent légèrement inférieurs à ceux obtenus par la suppression de la casse. Il est alors difficile de dire quelle méthode effectue la meilleure correspondance entre les formes textuelles et leurs concepts. Du point de vue recherche d'information, la suppression de la casse est plus simple à mettre en œuvre et plus rapide.

Globalement, la prise en compte des variations de casse améliore les résultats. Cette amélioration est nette dans le modèle utilisant toutes les formes textuelles où elle atteint 31%. Nous constatons aussi que la résolution de ces variations a moins d'impact sur la pondération *DFR*. En effet, cette pondération donne des poids élevés à certains concepts non pertinents. La prise en compte des variations de casse introduit du bruit sur les concepts, par exemple la forme textuelle *'a'* correspond par cette méthode au concept *'autopsie'*. Par l'utilisation de la pondération *DFR*, le poids affecté à ce concept est supérieur à celui affecté au concept correspondant à *'infarctus du myocarde',* concept correct et plus précis. Le filtrage d'une partie de ces mauvais concepts à l'aide des étiquettes syntaxiques améliore les résultats de 25% avec la pondération *DFR*. Au contraire, ce même filtrage a peu d'impact lors de l'utilisation du *tf.idf*.

Lors de ces expérimentations, nous remarquons la non détection, au niveau lexical, de termes qui pourraient être associés à des concepts. Ce type d'erreur provient de l'analyse lexicale de TreeTagger qui ne permet pas de retrouver les lemmes de tous les mots utilisés dans le corpus. Par exemple, le terme *'angiograms'*, présent dans une requête sous la forme plurielle, n'est pas associé au concept correspondant car UMLS ne contient que la forme singulière *'angiogram'* et car TreeTagger n'est pas capable de retrouver le lemme correspondant à *'angiograms'*. En effet, l'analyseur TreeTagger est un analyseur général, non adapté au vocabulaire médical. L'utilisation d'un analyseur spécialisé sur le domaine pourrait améliorer les résultats.

Les deux dernières méthodes proposées pour réduire l'ambiguïté impactent peu les résultats, même si l'étude des requêtes montre une meilleure détection des concepts après le filtrage de certains thésaurus.

Malgré une détection incomplète des concepts, le modèle conceptuel permet d'obtenir une amélioration par rapport au modèle à base de lemmes. Extraire les concepts qui correspondent à toutes les formes textuelles, permet d'obtenir une bonne couverture des concepts. Le peu de variation des termes dans la collection médicale a pour conséquence qu'obtenir les bons concepts n'a finalement pas autant d'impact que prévu si l'erreur d'assignation reste systématique. Cependant, cette erreur devient gênante lorsqu'on souhaite utiliser des informations sur ces concepts comme, par exemple, les relations. En particulier, l'assignation de la classe sémantique au concept peut devenir incorrecte.

2.4 Méthode MapMiniPar

La méthode de surface pour la détection des concepts peut se transposer à d'autres analyseurs. Nous utilisons MiniPar comme deuxième analyseur syntaxique servant de base à la détection des concepts. Pour cet analyseur, nous n'évaluons que les trois filtrages obtenus après suppression de la casse sur la méthode extrayant tous les concepts (cf. tableau 24). Globalement la méthode MapMiniPar donne des résultats similaires à ceux utilisant l'analyseur TreeTagger. Sur le *tf.idf* la meilleure méthode reste celle utilisant les filtres sur les étiquettes. Par contre sur la pondération *DFR*, les meilleurs résultats sont obtenus sans filtrage.

	sans casse	F1	F2	F3	Tf.idf	DFR
Toutes les formes textuelles	X				0.1556	**0.1820**
	X	X			**0.1559**	0.1809
	X	X	X		0.1502	0.1784
	X	X	X	X	0.1464	0.1782

Tableau 24 Résultats en précision moyenne obtenus sur les concepts à l'aide de l'extraction MapMiniPar sur A_IMG_05

2.5 Méthode MetaMap

Nous évaluons ensuite l'extraction des concepts par l'outil MetaMap, cet outil fournit plusieurs résultats :

- d'une part, un ensemble de concepts candidats, c'est-à-dire l'ensemble des concepts qui peuvent être mis en correspondance avec la totalité ou une partie d'un syntagme nominal,
- d'autre part, la liste des correspondances, c'est-à-dire l'ensemble des groupes de concepts couvrant l'ensemble du syntagme.

Dans ces deux ensembles, chaque élément possède une pondération. Le tableau 25 décrit l'utilisation de la meilleure correspondance détectée par MetaMap combinée à une sélection d'éléments candidats en fonction de leur variation par rapport au texte.

Candidat	Meilleure correspondance	Tf.idf	DFR
Sans variation	X	**0.1642**	**0.1975**
Sans variation		0.1557	0.1875
Avec variation	X	0.1475	0.1556
	X	0.1140	0.1148

Tableau 25 Résultat en précision moyenne de l'extraction des concepts par MetaMap

Les résultats montrent que l'utilisation seule de la meilleure correspondance ne suffit pas, cela donne les plus mauvais résultats en termes de précision moyenne sur les deux pondérations. En effet, comme avec l'utilisation de la méthode précédente, sélectionner seulement la meilleure correspondance reste trop restrictif et par conséquent le rappel diminue. Cela se voit sur la figure 50 où seul le premier point de rappel de la méthode utilisant seulement la correspondance est au dessus de la méthode ayant la meilleure précision moyenne. Nous remarquons ensuite qu'utiliser seulement les concepts dont les termes correspondent exactement au texte permet d'obtenir de bons résultats en recherche d'information ; ces résultats restent très proches de ceux obtenus par les méthodes précédentes. Cependant ils les améliorent par l'ajout des concepts qui constituent la meilleure correspondance. Enfin, augmenter le nombre de concepts candidats pris en compte sur le critère de la variabilité diminue les résultats. Ajouter des concepts correspondant à des termes ayant des formes trop variables rajoute trop de bruit et diminue les résultats.

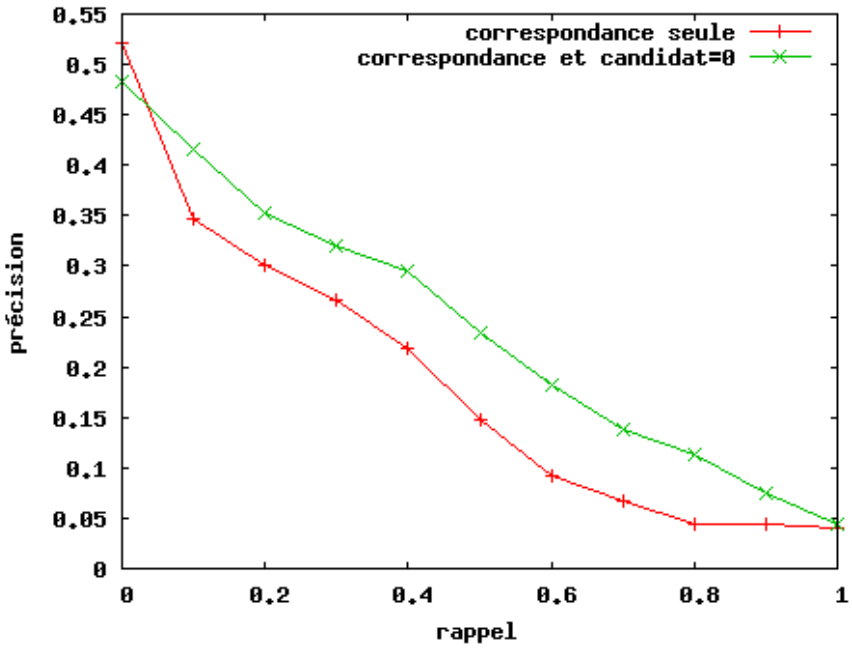


Figure 50 Courbe de rappel précision comparant la correspondance seule face à la correspondance associée aux candidats en tf.idf.

2.6 Synthèse

Par rapport au modèle de référence sur les lemmes, les trois méthodes permettant l'indexation par concepts fournissent des résultats équivalents voire supérieurs. L'utilisation de concepts permet donc de faire aussi bien voire d'améliorer des modèles à base de lemmes.

Les trois méthodes d'extraction des concepts donnent des résultats proches, elles possèdent cependant leurs spécificités. Au niveau global, la méthode MetaMap donne les meilleurs résultats, cela vient essentiellement du fait que MetaMap améliore plus le rappel que la précision, car cette méthode extrait plus de concepts. Sur la courbe de rappel précision (figure 51), la courbe correspondant à MetaMap dépasse les autres, essentiellement pour les points de rappel intermédiaires. Au contraire la méthode MapMiniPar donne la meilleure précision (cf. tableau 26) en *tf.idf*, cette méthode donne aussi la meilleure précision au premier point de rappel, c'est la méthode qui donne les concepts les plus précis. Enfin, la méthode MapTreeTagger a une courbe de rappel précision très proche de celle des lemmes. Elle donne la meilleure précision en *DFR*. La méthode de détection des concepts par analyse

morphosyntaxique donne donc les meilleurs précisions à 5 documents ; cette approche est donc intéressante pour la recherche d'information orientée précision.

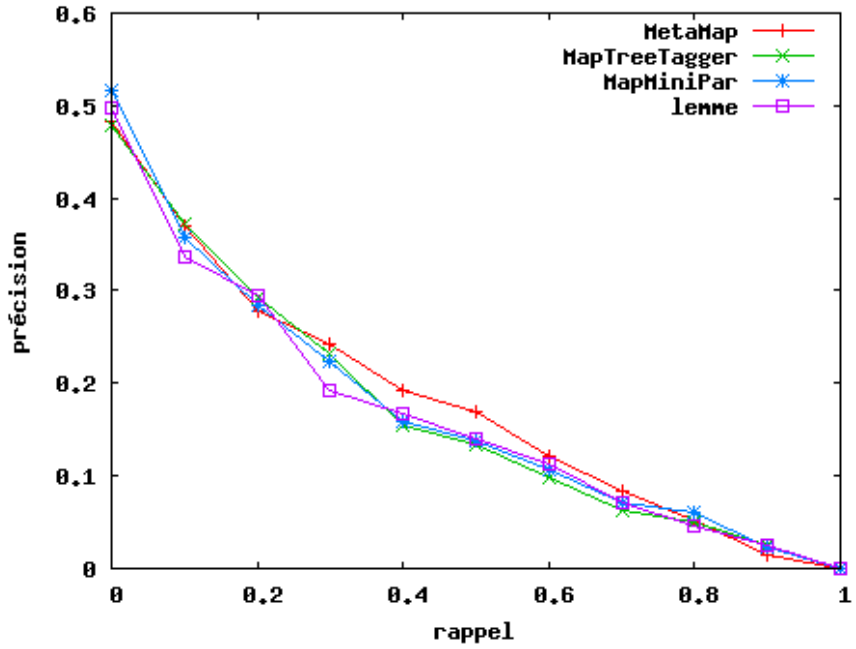


Figure 51 Courbe de rappel précision comparant les différentes extractions de concept et la méthode de référence en tf.idf.

	MAP		P@5	
	tf.idf	DFR	tf.idf	DFR
MapMiniPar	0.1559	0.1809	0.3040	0.4160
MapMiniPar + filtrages	0.1464	0.1782	**0.3120**	0.4400
MapTreeTagger	0,1566	0,1859	0.2880	0.3120
MapTreeTagger +filtrages	0,1541	0,1827	0.3040	**0.4480**
MetaMap	**0.1642**	**0.1975**	0.3040	0.4400

Tableau 26 Résultats en précision moyenne et en précision à 5 documents du modèle IC en fonction de la méthode de détection de concepts sur A_IMG_05

2.7 Extraction multilingue

L'utilisation de concepts permet d'avoir une interrogation multilingue des représentations de documents, nous évaluons ici une telle approche. Nous proposons trois évaluations différentes. Sur ces évaluations, la méthode MapTreeTagger permet l'indexation des parties française et allemande du corpus. Chaque évaluation fait ensuite varier l'indexation de la partie anglaise du corpus en utilisant chacune de nos trois méthodes de détection des concepts. A l'indexation, un document multilingue se décompose en plusieurs sous-documents, un pour chaque langue. Nous indexons chacune des langues séparément et, à l'interrogation, le score conservé pour un document multilingue correspond au meilleur score obtenu par l'un de ces sous-documents. Pour les requêtes, nous interrogeons toujours avec la requête anglaise, assortie de la méthode de production des concepts correspondant à l'anglais.

	MAP	P@5
MapTreeTagger	0.1447	0.2635
MapMiniPar	0.1391	0.2880
MetaMap	0.1661	0.2880

Tableau 27 Résultats de l'extraction des concepts sur la collection complète

Sur les résultats (cf. tableau 27), nous remarquons qu'utiliser les concepts sur la collection entière fournit des résultats inférieurs à l'utilisation des concepts sur la seule partie anglaise de la collection pour les méthodes à base d'analyseur morphosyntaxique. Les résultats restent stables pour MetaMap. Le niveau de ces résultats peut venir de la méthode d'obtention des résultats de CLEF 2005 (pool) où aucune méthode n'utilisait de concepts multilingues.

3 Détection des relations

La méthode utilisée pour extraire les relations reste la même quelle que soit l'extraction des concepts utilisée, nous comparons ici le résultat obtenu pour l'utilisation seule des relations sur les trois extractions de concepts.

3.1 Méthode

Nous évaluons ici l'extraction des relations à l'aide d'un modèle relationnel *IR*. Ce modèle se base sur des relations sémantiques. Ces relations sont détectées à partir des concepts produits par l'une des fonctions de détection des concepts. Ce modèle utilise un vocabulaire pondéré calculé soit par le *tf.idf*, soit par le *DFR*. Le modèle s'écrit alors :

$$IR = \left(ST_{UMLS}, SVD_{IR}, SVQ_{IR}, RC_{IR}\right)$$

$$SVD_{IR} = SVQ_{IR} = \left(V_{relations}^{IR}\right) \text{ avec } V_{relations}^{IR} \subseteq V_{relations} \times P^1$$

$DV_{relations}^{IR} \subset V_{relations}^{IR}$ constitue l'ensemble des relations détectées dans les phrases du document et P^1 le *tf.idf* ou la *DFR*

$QV_{relations}^{IR} \subset V_{relations}^{IR}$ constitue l'ensemble des relations détectées dans la requête et P^1 est leur fréquence.

RC_{IR} utilise le produit scalaire.

3.2 Résultats

Nous évaluons les résultats obtenus par l'utilisation des relations détectées sur les trois méthodes de détection des concepts. Le tableau 28 présente les résultats en précision moyenne et en précision à 5 documents.

	Nombre de requêtes	MAP		P@5	
		tf.idf	DFR	tf.idf	DFR
MapMiniPar	19	0.1236	0.1336	0.2947	0.3895
MapMiniPar + filtrages	20	0.1228	0.1286	**0.3400**	**0.4300**
MapTreeTagger	21	0.1315	0.1255	0.3238	0.3333
MapTreeTagger +filtrages	21	0.1331	0.1407	0.3238	0.4095
MetaMap	25	0.1377	0.1608	0.2480	0.3520

Tableau 28 Résultats en précision moyenne et en précision à 5 documents de la détection des relations selon la méthode de détection de concepts sur A_IMG_05

Ces résultats montrent, premièrement, que les relations seules donnent des résultats inférieurs aux concepts seuls et aux lemmes. Nous remarquons que, par l'utilisation des relations seules, la méthode MetaMap résout toutes les requêtes alors que la méthode MapMiniPar ne répond qu'à 19 requêtes. Nous n'effectuons que des correspondances exactes entre les relations de la requête et celles des documents. Par conséquent, le système ne retrouve pas les documents où les concepts ne sont pas reliés.

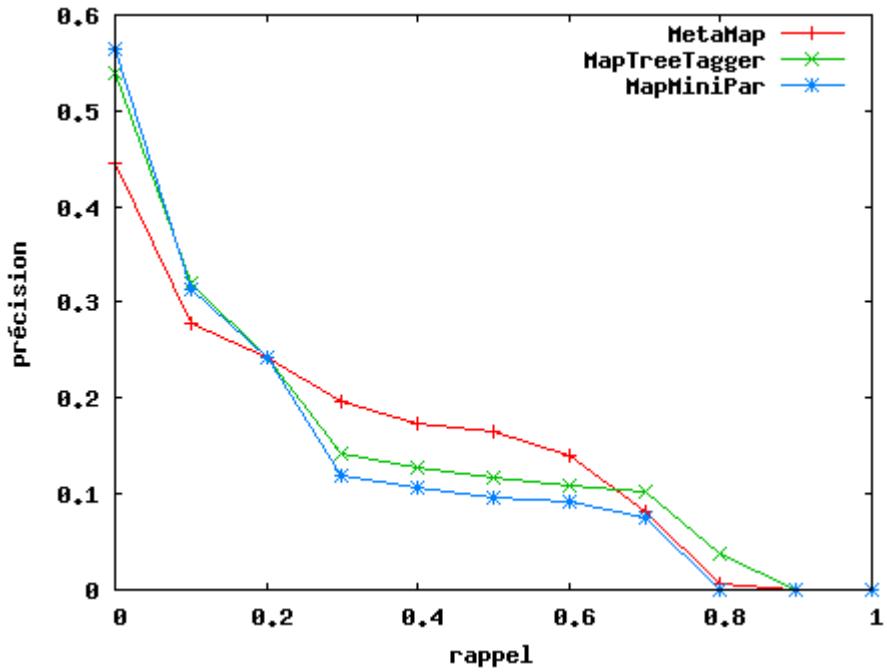


Figure 52 Courbe de rappel précision comparant les trois extractions de relations (MapMiniPar+filtrage, MapTreeTagger+filtrages, MetaMap) en tf.idf.

La courbe de rappel précision de la figure 52 présente les résultats des trois méthodes d'extraction des relations. Comme pour les concepts, ces méthodes fournissent des comportements différents. Les deux méthodes MapTreeTagger et MapMiniPar avec filtrages donnent de meilleurs résultats sur les premiers points de rappel alors qu'avec MetaMap, les meilleurs points de rappel constituent les points intermédiaires.

La méthode MapMiniPar obtient la meilleure précision à 5 documents (tableau 28) ; en effet si avec les relations cette méthode ne permet pas de répondre à toutes les requêtes, cette méthode détecte les relations les plus adéquates. De plus, sur cette méthode comme sur la méthode MapTreeTagger, l'ajout

des filtrages améliore les résultats. Détecter les bons concepts et les bonnes relations améliore la précision à 5 documents, cela est donc important pour répondre à des besoins experts.

3.3 Synthèse

La méthode de détection des concepts MetaMap fournit de nombreux concepts, avec moins de précision que les deux autres méthodes. Par conséquent cette méthode détecte plus de relations dans les phrases. Ces relations couvrent plus de concepts et permettent donc de répondre à toutes les requêtes. Cependant ces relations demeurent beaucoup plus diverses que celles sélectionnées par les deux autres méthodes, et de ce fait, si cette méthode donne les meilleurs résultats en terme de précision moyenne, elle ne donne pas les meilleurs résultats en précision à 5 documents.

4 Bilan

Nous évaluons ici trois méthodes automatiques qui permettent d'extraire les concepts. Sur ces trois méthodes, nous testons un modèle de recherche d'information conceptuel et un modèle relationnel. Les trois approches montrent des comportements différents, les approches à base d'analyseur syntaxique donnent de meilleures précisions alors que l'approche MetaMap donne un meilleur rappel. Globalement les approches à base de concepts donnent des résultats en précision moyenne légèrement supérieurs aux résultats obtenus par les lemmes. Ces expérimentations prouvent l'intérêt des indexations conceptuelles. Les méthodes de détection des concepts se basent sur des méthodes simples qui fournissent une surreprésentation du contenu des documents. Les expérimentations montrent donc que ces méthodes sont valables et qu'elles permettent d'obtenir des indexations conceptuelles automatiques qui rivalisent avec des indexations à base de mots-clefs, ce que peu de travaux avaient réussit à montrer jusqu'alors.

Les modèles relationnels quant à eux donnent des résultats inférieurs en précision moyenne par rapport aux approches à base de lemmes ; de plus ils ne permettent pas de résoudre toutes les requêtes. Cependant, dans certains cas, ils donnent des résultats de précision à 5 documents proches voire supérieurs à ceux des concepts.

Dans la mesure où ces trois méthodes montrent des comportements différents, nous pourrons utiliser l'une ou plusieurs d'entre elles dans la suite de nos expérimentations. Nous sélectionnons pour cela les paramètres suivants pour ces extractions :

- **MapTreeTagger** : détecte les concepts résultants de tous les filtrages
- **MapMiniPar** : détecte les concepts résultants de tous les filtrages
- **MetaMap** : détecte les concepts de la meilleure correspondance et les concepts candidats sans variation.

Chapitre XI Modèle Local ML

« Le vrai génie réside dans l'aptitude à évaluer l'incertain, le hasardeux, les informations conflictuelles. » Winston Churchill

Ce chapitre présente l'évaluation du modèle local sur la collection CLEF Médicale. Le modèle local se base sur des pondérations à base de *tf.idf ;* en effet ce modèle combine des informations provenant des relations et des concepts. Combiner de telles informations peut se faire dans des modèles simples, par exemple le modèle vectoriel à base de *tf.idf*. Combiner linéairement des pondérations telles que la *divergence from randomness* (*DFR)* n'a pas de sens, dans ce cas nous devrions modifier le calcul de la pondération pour prendre en compte les relations au sein même de ce calcul.

Nous comparons dans ce chapitre le modèle local *ML* avec deux modèles, l'un qui contient seulement des concepts, l'autre seulement des relations. Nous souhaitons démontrer que l'utilisation de plusieurs points de vue permet d'améliorer la précision des résultats. Ainsi nous comparons le modèle local avec des modélisations conceptuelles, relationnelles ou à base de lemmes, ces dernières se basant aussi sur des variations du *tf.idf*. Nous évaluons ces modèles en testant différentes variations du *tf.idf* :

- (*occ*) la fréquence du concept (ou de la relation) au sein d'un document,
- (*tf*) le log de la fréquence défini par :

$$tf = log(n_i) - 1$$

  avec n_i la fréquence du concept ou de la relation dans le document,

- (*idf*) la fréquence documentaire inverse calculée par :

$$idf = log\left(\frac{N}{d_i}\right)$$

  avec d_i le nombre de documents contenant le concept, et N le nombre de documents de la collection,

- (*tf.idf*) qui utilise ensemble le *tf* et l'*idf.*

Nous évaluons ce modèle sur la partie anglaise du corpus, d'une part en l'évaluant au niveau des images A_IMG_05 (collection 2005, toutes les langues et évaluation au niveau des images) et d'autre part au niveau des diagnostics en anglais EN_DIAG_05 (collection 2005, anglais seulement et évaluation au niveau des images).

1 Mise en œuvre

La modélisation des documents selon le modèle local à partir de la représentation intermédiaire se base sur une adaptation de méthodes vectorielles existantes. Cette adaptation est simple, elle permet une indexation rapide et fournit des temps de réponse très courts à l'interrogation. Nous avons aussi implémenté le calcul des scores de correspondance sur les relations. Pour cela nous avons défini de nouvelles méthodes permettant ces calculs.

La programmation des méthodes de calcul des scores de confiance et l'expérimentation du modèle local représentent environ trois mois de travail.

2 Méthode de référence

Nous utilisons toujours la méthode de référence basée sur le modèle à base de lemmes *IL* (présenté dans le chapitre précédent). Cependant dans cette partie nous utilisons le modèle *IL* à l'aide de plusieurs variations du *tf.idf*. Nous présentons les résultats de cette méthode de référence pour la partie anglaise du corpus dans le tableau 29.

Les résultats montrent que la pondération *occ* donne les meilleurs résultats. Cependant tous les résultats varient dans un intervalle de moins de 1%. Nous notons enfin que ces résultats restent faibles.

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf. Idf	0.1543	**0.3760**	0.1806	0.3760
Idf	0.1565	**0.3760**	0.1834	0.3600
Tf	0.1659	0.3680	0.1920	0.3680
occ	**0.1661**	0.3680	**0.1942**	**0.3680**

Tableau 29 Précision moyenne et précision à 5 documents pour le modèle à base de lemmes

3 MapMiniPar

Dans cette partie nous évaluons les résultats obtenus par le modèle local avec la production des graphes à l'aide de la méthode MapMiniPar. Cette méthode donne les meilleurs résultats en précision à 5 documents sur le modèle conceptuel *IC*.

Nous comparons le modèle *ML* avec un modèle conceptuel *IC* et un modèle relationnel *IR* utilisant des variations du *tf.idf*. Nous notons que ces deux modèles correspondent à deux instances particulières du modèle *ML* pour lesquelles le modèle ne prend pas en compte respectivement les relations ou les concepts.

Après avoir présenté les résultats des concepts et des relations, nous comparons ces résultats à ceux obtenus par le modèle local sans utilisation de score de confiance. Nous testons ensuite l'ajout du score de confiance au sein du modèle local.

3.1 Concept et relation

Nous présentons dans le tableau 36 les résultats du modèle conceptuel obtenus sur les variations du *tf.idf* et nous faisons de même dans le tableau 37 pour le modèle relationnel.

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf.Idf	0.1464	0.312	0.2122	0.336
Idf	0.1507	0.336	0.2204	0.36
Tf	0.1649	0.384	0.2363	0.4
occ	**0.1672**	**0.408**	**0.2461**	**0.424**

Tableau 30 Résultats des concepts détectés par la méthode MapMiniPar

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf.Idf	0.1228	0.34	0.2053	0.32
Idf	0.1239	**0.36**	**0.2056**	**0.34**
Tf	0.1225	0.34	**0.2056**	0.33
Occ	**0.1243**	**0.36**	0.2055	**0.34**

Tableau 31 Résultats des relations sur la détection des concepts MapMiniPar

Le modèle à base de concepts donne de meilleurs résultats que le modèle à base de relations. L'utilisation des relations ne résout que 20 requêtes sur les 25 de CLEF 2005. Les cinq requêtes restantes ne contiennent pas de relations, ou les relations détectées ne possèdent pas d'équivalent dans le corpus des documents. Les relations seules sont trop restrictives ; peu de documents possèdent toutes les relations de la requête. De plus, le modèle relationnel ne retrouve pas les concepts composant les relations. Les résultats montrent aussi le faible impact des variations de pondération sur les résultats des relations.

Les résultats du modèle conceptuel se rapprochent de ceux obtenus par le modèle à base de lemmes dans l'évaluation au niveau des images, mais donnent de meilleurs résultats sur l'évaluation du texte en anglais EN_DIAG_05. Les concepts s'avèrent meilleurs que les lemmes pour détecter directement les diagnostics.

3.2 Modèle Local sans confiance

Nous présentons dans le tableau 32 les résultats obtenus par le modèle local sur la collection à l'aide de l'extraction des concepts par MapMiniPar, cela sans utiliser de score de confiance.

		A_IMG_05		EN_DIAG_05	
concept	relation	MAP	P@5	MAP	P@5
Occ	Tf	0.1704	**0.4**	0.2536	0.384
	Idf	0.1714	**0.4**	0.2551	0.384
	Occ	0.1715	0.376	0.2551	**0.392**
Tf	Tf	0.1698	0.392	0.2535	0.384
	Idf	0.1714	**0.4**	0.2550	**0.392**
	occ	**0.1718**	**0.4**	**0.2561**	**0.392**

Tableau 32 Précision moyenne et précision à 5 documents du modèle local appliqué a l'extraction des concepts MapMiniPar

Les résultats montrent que la combinaison des relations et des concepts dans le modèle permet d'améliorer les résultats en précision moyenne de 2,7%, par rapport aux concepts, sur A_IMG_05 et de 4% sur E_DIAG_05. L'amélioration augmente avec la sélection des résultats au niveau diagnostic.

Le modèle permet d'améliorer la sélection des diagnostics même si l'impact de cette amélioration se voit moins au niveau des images. Par contre, la précision à 5 documents reste la même.

3.3 Modèle local avec confiance sur les relations

Ici, pour chaque relation sémantique, nous prenons en compte le score de confiance $p_{confiance}$. Nous calculons ce score à l'aide de la collection et des deux méthodes unigramme et bigramme présentées dans la section 3.1.2 du chapitre IX. Nous évaluons l'impact de ces deux méthodes sur le modèle relationnel *IR* et sur le modèle local *ML*. Nous calculons le score de recherche d'information p_{ri} à l'aide d'un *tf*. Le tableau 33 et le tableau 34 présentent les résultats obtenus par nos méthodes de calcul de confiance.

	MAP	P@5
IR	0.2185	0.4300
ML	0.2635	0.4480

Tableau 33 Résultats avec confiance selon le modèle unigramme en tf.idf

	ML		IR	
λ	MAP	P@5	MAP	P@5
0	0.2635	0.4480	0.2185	0.4300
0.2	0.2612	0.4320	0.2107	0.4000
0.5	0.2590	0.4320	0.2139	0.4100
0.7	0.2567	0.4400	0.2133	0.4000
1	0.2515	0.4240	0.2083	0.3900

Tableau 34 Résultats avec confiance selon le modèle bigramme avec λ paramètre de lissage entre la probabilité unigramme et la probabilité bigramme

Les résultats montrent que l'utilisation de la confiance améliore la précision moyenne du modèle local d'environ 3,9 %. Cette amélioration provient de l'amélioration de la précision à 5 documents sur les relations qui passent de 0,33 à 0,43. Par comparaison aux concepts, les résultats obtenus avec la confiance fournissent une amélioration de la précision moyenne. Ils montrent aussi une augmentation de la précision à 5 documents, ce que la non-utilisation de la confiance ne permet pas. Ces résultats montrent l'intérêt de ce score dans un modèle orienté précision. La comparaison des deux modèles montre que le modèle bigramme est moins efficace que le modèle unigramme. Notre corpus qui sert aussi d'apprentissage n'est pas assez grand et ses relations demeurent trop ambigües pour que les bigrammes soient discriminants.

3.4 Synthèse

Sur le modèle local, qui combine les concepts et les relations, la précision moyenne est supérieure à celle obtenue par les concepts ou par les relations seules, quel que soit le poids. Sur l'évaluation A_IMG_05, le meilleur modèle *ML* améliore de 4% les résultats par rapport au meilleur modèle à base de concepts et de 4.7% par rapport au meilleur modèle à base de lemmes. Sur l'évaluation EN_DIAG_05, le modèle *ML* améliore de 7% les performances par rapport aux modèles à base de concepts et les améliore d'environ 36% par rapport aux modèles à base de lemmes. Le tableau 35 synthétise ces résultats.

Par conséquent, le modèle *ML* offre de meilleures performances que les concepts ou les relations seuls. Notons également que le modèle *ML* améliore la précision à 5 documents. Cette amélioration

n'intervient cependant que lorsque le modèle utilise la confiance, 4% d'amélioration sur A_IMG_05 et 7.5% sur EN_DIAG_05. Cela ne s'avère pas surprenant car la confiance discrimine les relations correctement extraites quand l'utilisation seule du score de recherche d'information ne permet pas de distinguer ces relations. Cela a pour conséquence de mieux sélectionner les premières réponses du système. Enfin nous remarquons que l'utilisation d'un score de confiance en complément d'un score de recherche d'information est pertinent dans le cadre de systèmes orientés précision.

		A_IMG_05		EN_DIAG_05	
concept	relation	MAP	P@5	MAP	P@5
Occ	Tf	0.1704	0.4	0.2536	0.384
	Occ	0.1715	0.376	0.2551	0.392
	Tf + confiance	0.1733	0.4160	0.2614	**0.448**
	Occ + confiance	0.1718	**0.424**	0.2600	**0.448**
Tf	Tf	0.1698	0.392	0.2535	0.384
	occ	0.1718	0.4	0.2561	0.392
	Tf + confiance	**0.1740**	0.4160	**0.2635**	**0.448**
	Occ + confiance	0.1724	**0.424**	0.2616	**0.448**

Tableau 35 Synthèse sur le modèle local utilisant la détection MapMiniPar en précision moyenne et précision à 5 documents

4 MetaMap

Dans cette partie nous évaluons les résultats obtenus par le modèle local avec la production des graphes à l'aide de la méthode MetaMap. Sur l'évaluation des méthodes d'extraction des concepts cette méthode donne les meilleurs résultats en précision moyenne. Comme dans la section précédente, nous comparons le modèle *ML* avec un modèle *IC* et un modèle *IR* en utilisant les variations du *tf.idf*.

Nous présentons les résultats des concepts et des relations dans le tableau 36 et le tableau 37. Nous comparons ensuite ces résultats aux résultats obtenus par le modèle local sans score de confiance. Nous testons enfin l'apport du score de confiance sur les concepts au sein de ce modèle.

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf.Idf	0.1642	0.3040	0.2350	0.3280
Idf	**0.1719**	**0.3120**	**0.2429**	**0.3360**
Tf	0.1580	0.3120	0.2018	0.3360
Occ	0.1642	0.3280	0.2020	0.3440

Tableau 36 Résultats des concepts détectés par la méthode MetaMap

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf.Idf	0.1377	0.2480	0.1778	0.2800
Idf	0.1426	0.2720	0.1819	0.2960
Tf	0.1458	0.2560	0.1868	0.2880
Occ	**0.1487**	**0.3040**	**0.1879**	**0.3200**

Tableau 37 Résultats des relations sur la détection des concepts MetaMap

Comme illustré dans le chapitre précédent, cette méthode fournit une précision à 5 documents plus faible que la méthode MapMiniPar et une précision moyenne plus élevée. Les meilleurs résultats sur les concepts sont obtenus avec la pondération *idf*. Cette méthode permet d'extraire de nombreux concepts, l'importance de l'*idf* augmente donc pour cette méthode car cette pondération permet de sélectionner les concepts les plus pertinents.

Pour les relations, cette méthode obtient de meilleurs résultats que ceux obtenus à l'aide de MapMiniPar au niveau des images. Ils sont cependant moins bons au niveau des diagnostics. Cela montre d'une part l'importance d'évaluer les résultats au niveau diagnostic car les résultats au niveau image ne reflètent pas obligatoirement la bonne sélection des diagnostics. D'autre part, cela montre que les relations extraites par MetaMap ne discriminent pas assez les documents pertinents.

4.1 Modèle Local sans confiance

Nous présentons dans le tableau 38 les résultats obtenus par le modèle local sur la collection à l'aide de l'extraction des concepts par MetaMap et sans utiliser de score de confiance.

		A_IMG_05		EN_DIAG_05	
concept	relation	MAP	P@5	MAP	P@5
Tf.idf	Tf	0.1637	0.2800	0.2064	0.2800
	Occ	0.1666	0.3200	0.2058	**0.3440**
Idf	Tf	0.1642	0.2720	0.2078	0.2880
	occ	**0.1685**	**0.3280**	**0.2079**	**0.3520**

Tableau 38 Précision moyenne et précision à 5 documents du modèle local appliqué à la détection des concepts MetaMap

Le regroupement des concepts et des relations au sein du modèle local a pour conséquence de diminuer les résultats. Les relations détectées dans cette méthode sont trop nombreuses et peu précises, elles n'apportent pas d'informations discriminantes.

4.2 Modèle Local avec confiance sur les concepts

MetaMap fournit un score sur la détection de concepts, nous utilisons ce score pour représenter la confiance sur les concepts. Nous évaluons ce score directement sur la modélisation à base de concepts *IC* (tableau 39).

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf.Idf	0.1663	0.2880	0.2361	0.3120
Idf	**0.1722**	**0.3200**	**0.2446**	**0.3440**
Occ	0.1585	0.3040	0.2013	0.3200

Tableau 39 Précision moyenne et précision à 5 documents des concepts détectés par la méthode MetaMap avec l'utilisation du score de confiance

L'utilisation du score de confiance sur les concepts de MetaMap n'a que très peu d'impact, cependant cet impact s'avère dans la majorité des cas positif. La raison pour laquelle cette amélioration reste si faible vient de la méthode utilisée pour déterminer le score de confiance. MetaMap fournit un score entre 0 et 1000 que nous transposons entre 0 et 1, cependant ce score ne

consiste pas en une probabilité. Les concepts que nous sélectionnons ne se répartissent pas uniformément entre 0 et 1. La majorité des concepts obtiennent un score entre 500 et 1000, et beaucoup obtiennent un score de 1000. Par conséquent, cette normalisation du score ne différencie pas suffisamment les concepts.

5 MapTreeTagger

Dans cette partie nous évaluons les résultats obtenus par le modèle local sur les graphes obtenus à l'aide de la méthode MapTreeTagger. Comme pour les deux méthodes de détection précédentes, nous détaillons les résultats sur des variations du *tf.idf* à l'aide du modèle conceptuel, du modèle relationnel et du modèle local. Le tableau 40, le tableau 41 et le tableau 42 présentent ces résultats.

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf.Idf	0.1521	0.3040	0.2180	0.3200
Idf	0.1571	0.3360	0.2265	0.3520
Tf	**0.1688**	0.3680	0.2325	0.3600
occ	0.1687	**0.3760**	**0.2342**	**0.3760**

Tableau 40 Résultats des concepts détectés par la méthode TreeTagger

	A_IMG_05		EN_DIAG_05	
	MAP	P@5	MAP	P@5
Tf.Idf	0.1331	0.3238	**0.2179**	0.3429
Idf	0.1341	0.3238	0.2177	0.3429
Tf	0.1341	0.3238	0.2178	0.3333
occ	**0.1336**	**0.3238**	0.2175	**0.3429**

Tableau 41 Résultats des relations sur la détection des concepts TreeTagger

		A_IMG_05		EN_DIAG_05	
concept	relation	MAP	P@5	MAP	P@5
Tf	Tf.Idf	**0.1770**	**0.3920**	**0.2491**	0.3920
	Occ	0.1751	0.3600	0.2469	0.3920
occ	Tf.idf	0.1758	0.3840	0.2476	0.3920
	occ	0.1743	0.3600	0.2458	**0.4000**

Tableau 42 Précision moyenne et précision à 5 documents du modèle local appliqué à la détection des concepts MapTreeTagger

Les résultats présentés ici montrent que pour cette méthode l'utilisation du modèle local est meilleure que l'utilisation des concepts ou des relations seuls. Dans cette méthode, la précision à 5 documents augmente par l'utilisation des relations sans utilisation de score de confiance, contrairement aux autres méthodes. Cette amélioration montre la complémentarité de ces deux vocabulaires. Cependant cette méthode ne permet pas d'obtenir de score de confiance.

6 Conclusion

Nous avons évalué notre modèle local sur les trois méthodes d'extraction pour les concepts, nous résumons les meilleurs résultats de ce modèle dans le tableau 43. Ces expérimentations montrent

l'intérêt d'utiliser dans un même modèle les informations conceptuelles et relationnelles, notamment pour améliorer la précision. Cependant améliorer les résultats nécessite que les relations soient précises. Si elles sont trop nombreuses et peu précises elles ne représentent alors plus que des informations de cooccurrence qui n'apportent pas d'amélioration, c'est le cas dans la méthode MetaMap.

			A_IMG_05		EN_DIAG_05	
	concept	relation	MAP	P@5	MAP	P@5
MapMiniPar	Tf	Tf + confiance	0.1740	0.4160	**0.2635**	**0.448**
	Tf	Idf + confiance	0.1730	**0.424**	0.2622	**0.448**
MetaMap	Idf + confiance	**aucune**	0.1719	0.3120	0.2429	0.3360
MapTreeTagger	Tf	Tf.Idf	**0.1770**	0.3920	0.2491	0.3920
	occ	occ	0.1743	0.3600	0.2458	0.4000

Tableau 43 Résumé des meilleures indexations ML pour les trois méthodes de détection des concepts

Au niveau des relations, nous remarquons que les variations de pondérations impactent très peu le résultat des relations. Les pondérations en *tf.idf* ne permettent pas de discriminer les relations. L'apparition d'une relation est une information suffisante quand elle se révèle juste.

Les expérimentations qui prennent en compte les scores de confiance montrent l'intérêt de ce score en recherche d'information. En effet dans le tableau 43 les meilleurs résultats sont obtenus par l'utilisation d'un score de confiance, quand celui-ci peut être calculé. Utiliser des informations produites par le traitement de la langue pour construire la représentation des documents montre donc un intérêt certain.

Le modèle local reste assez simple, il se base sur des pondérations du modèle vectoriel. Si ce modèle considère les relations et les concepts comme différents, la pondération des relations à base de *tf.idf* ne prend pas en compte correctement les spécificités des relations.

Ces expérimentations montrent d'une part l'intérêt de l'utilisation de représentations expressives, et d'autre part la faisabilité d'une indexation de niveau sémantique et de son automatisation sur un domaine précis. Cependant les résultats obtenus par cette méthode, même par l'intégration des relations et des scores de confiance, restent inférieurs à des méthodes de pondération plus évoluées telles que la pondération *DFR*. Dans ces modèles de pondération, les relations doivent s'intégrer directement dans le modèle de correspondance et ne peuvent pas utiliser une simple combinaison linéaire. C'est ce que nous faisons dans le chapitre suivant à l'aide du modèle global.

Chapitre XII Modèle Global MG

« Idéalement nous sommes ce que nous pensons. Dans la réalité, nous sommes ce que nous accomplissons. »
Ayrton Senna

Ce chapitre évalue le modèle global sur la collection CLEF Médicale. Nous évaluons plusieurs variations de ce modèle, celles-ci portent sur le calcul de la probabilité de génération du graphe de la requête. Les différents modèles de graphe contiennent un certain nombre de paramètres. Dans les expérimentations suivantes nous apprenons ces paramètres sur une partie de la collection puis nous les évaluons sur une autre.

Ce chapitre présente, dans un premier temps, un modèle de référence basé sur les concepts. Il compare ensuite ce modèle à un premier modèle de graphe qui ne tient pas compte des étiquettes associées aux couples de concepts dans les graphes ; cela permet de nous situer par rapport à l'état de l'art.

Dans un deuxième temps, une section compare le modèle basé sur les concepts à des modèles de graphes complets. Nous étudions deux variations de ce modèle : la première n'utilise qu'une seule probabilité, la deuxième décompose cette probabilité en deux composantes.

Nous proposons enfin quelques expérimentations complémentaires et nous décrivons une indexation complète et multilingue prenant en compte les différentes méthodes d'extraction de concepts. Cette méthode, soumise à la campagne CLEF 2007 comme participation du laboratoire LIG, nous a permis d'obtenir les meilleurs résultats lors de cette campagne parmi les 147 participations de 13 groupes internationaux.

1 Mise en œuvre

La modélisation des documents selon le modèle global à partir de la représentation intermédiaire a nécessité la mise en place d'un système complet. Ce système se base sur des fichiers inverses pour accélérer l'interrogation. Le temps d'indexation à partir des représentations intermédiaires est d'environ une demi-heure pour la collection. Le temps d'exécution d'une requête est quant à lui en moyenne inférieur à 5 secondes.

Pour le modèle global nous avons implémenté différentes variations du modèle mais nous avons aussi implémenté le modèle graphique de (Gao *et al.*, 2004) auquel nous nous comparons.

La programmation du modèle et son expérimentation représentent environ quatre mois de travail.

2 Méthode de référence

Un modèle de langue unigramme sur les concepts permet d'évaluer les performances du modèle global. Ce modèle considère la requête comme un ensemble de concepts *C* et il évalue la probabilité que le modèle du document M_D génère cet ensemble de concepts.

$$P(C|M_D)$$

Comme pour le modèle global, nous émettons l'hypothèse que les concepts se génèrent indépendamment les uns des autres. Nous calculons la probabilité d'un concept à l'aide d'un lissage de Jelinek-Mercer entre cette probabilité calculée sur la collection et cette même probabilité calculée sur le document :

$$P(C|M_D) = \prod_{c \in C} P(c|M_D^g)$$

$$\text{où } P(c|M_D) = (1-\lambda_{concept})\frac{D(c)}{D(*)} + \lambda_{concept}\frac{C(c)}{C(*)}$$

Le modèle d'indexation unigramme sur les concepts que nous nommons *MLC* s'écrit alors :

$$MLC = (ST_{UMLS}, SVD_{MLC}, SVQ_{MLC}, RC_{MLC})$$

$$SVD_{MLC} = V_{concepts}^{MLC} \text{ avec } V_{concepts}^{MLC} \subseteq V_{concepts} \times P^1$$

$$SVQ_{MLC} = V_{concepts}$$

$DV_{concepts}^{MLC} \subset V_{concepts}^{MLC}$ avec $f_{np}(DV_{concepts}^{MLC}) = f_{np}(V_{concepts}^{MLC}) = V_{concepts}$ et $P^1 = P(c|M_D) = P(uv_{concept}|M_D)$

$QV_{concepts}^{MLC} \subset V_{concepts}$ constitue l'ensemble des concepts détectés dans la requête

RC_{MLC} utilise la vraisemblance de la requête.

Nous notons que ce modèle équivaut à un modèle global qui ne prend pas en compte les relations.

3 Modèle global sans étiquette

3.1 Définition

Dans cette partie nous testons notre modèle global sans prendre en compte les étiquettes. Pour cela, nous considérons que deux concepts d'une phrase se trouvent soit en relation soit en non relation, le modèle ne considère alors que deux étiquettes 0 et 1 définies selon :

- $type(c1,c2)=1$, si *c1* et *c2* sont reliés,
- $type(c1,c2)=0$, si *c1* et *c2* ne sont pas reliés.

Nous utilisons ces étiquettes dans le modèle de graphe *MG*. Nous calculons les estimations du modèle selon deux méthodes. Dans une première, le nombre de relations $C(uv'_{relation} = (uv'_{concept1}, uv'_{concept2}, 0))$ et $D(uv'_{relation} = (uv'_{concept1}, uv'_{concept2}, 0))$ se décompte sur la collection et sur le document. Cependant du fait que le système détecte la relation dans une phrase, si

elle existe dans UMLS, ces décomptes sont égaux au décompte du nombre de couples $C(uv'_{couple} = (uv'_{concept1}, uv'_{concept2}))$ et $D(uv'_{couple} = (uv'_{concept1}, uv'_{concept2}))$ par conséquent :

$$\frac{C(uv'_{relation})}{C(uv'_{couple})} = \frac{D(uv'_{relation})}{D(uv'_{couple})} = 1$$

si $uv'_{concept1}$ et $uv'_{concept2}$ apparaissent ensemble dans au moins une phrase de la collection ou du document et 0 sinon.

Au final, la probabilité dans un document représente juste le rapport entre le lissage et 0 ou 1, nous nommons cette méthode la méthode **brute** :

- $P(uv'_{relation} | M_D^g) = 1$ si $uv'_{concept1}$ et $uv'_{concept2}$ apparaissent dans une même phrase du document,
- $P(uv'_{relation} | M_D^g) = \lambda_{relation}$ si $uv'_{concept1}$ et $uv'_{concept2}$ n'apparaissent pas dans une même phrase du document.

Pour obtenir une probabilité plus variable, nous proposons de calculer $C(uv'_{concept1}, uv'_{concept2})$ et $D(uv'_{concept1}, uv'_{concept2})$ par le nombre de fois où $uv_{concept1}$ ou $uv_{concept2}$ apparaissent dans une phrase de la collection, nous nommons cette méthode la méthode **aménagée**. Elle se calcule par:

$$C(uv'_{relation} = (uv'_{concept1}, uv'_{concept2}, 0)) = C(uv'_{concept1}, uv'_{concept2}) - C(uv'_{relation} = (uv'_{concept1}, uv'_{concept2}, 1))$$

Idem pour le document

3.2 Évaluation

Dans cette expérimentation, nous divisons les requêtes de CLEF médicale 2005 et 2006 en deux ensembles de requêtes : 25 requêtes[23] sélectionnées aléatoirement pour l'apprentissage et 30[24] pour tester les résultats de recherche d'information. Nous comparons de plus le modèle proposé avec le modèle proposé par GAO (Gao *et al.*, 2004) détaillé dans l'état de l'art. Ce modèle s'applique sur des arbres de dépendance, nous l'appliquons ici sur des graphes de concepts bien qu'en principe il ne puisse pas s'appliquer sur des graphes avec cycle. Nous obtenons les résultats présentés sur le tableau 44 pour la précision moyenne et les résultats présentés sur le tableau 45 pour la précision à 5 documents[25]. Nous ne présentons que les résultats obtenus à l'aide de la méthode d'extraction des concepts MapMiniPar. Nous fournissons dans l'annexe B les résultats de ces modèles obtenus sur une analyse syntaxique des documents et des requêtes du corpus de CLEF médicale.

[23] Requêtes 1 4 7 10 11 13 15 18 22 24 de 2005 et 2 4 8 9 11 13 14 17 19 20 23 25 26 27 30 de 2006

[24] Requêtes 2 4 8 9 11 13 14 17 19 20 23 25 26 27 30 de 2005 et 1 3 5 6 7 10 12 15 16 18 21 22 24 28 29 de 2006

[25] Pour la sélection des paramètres notamment pour la P5D en cas d'égalité, le paramètre le plus faible est sélectionné

	$\lambda_{concept}$	$\lambda_{relation}$	entraînement	évaluation
GAO	0.1	0.9	0.256	0.333
MG brut	0.1	0.3	0.256	**0.343**
MG aménagé	0.1	0.8	0.243	0.337
MLC	0.1		0.255	0.339

Tableau 44 Résultats en précision moyenne pour le modèle global sans étiquette sur la détection des concepts MapMiniPar, avec $\lambda_{concept}$ et $\lambda_{relation}$ comme paramètre de lissage

	$\lambda_{concept}$	$\lambda_{relation}$	entraînement	évaluation
GAO	0.1	0.9	0.450	0.440
MG brut	0.1	0.1	0.433	**0.480**
MG aménagé	0.1	0.1	0.433	**0.480**
MLC	0.1		0.408	0.453

Tableau 45 Résultats en précision à 5 documents pour le modèle global sans étiquette sur la détection des concepts MapMiniPar, avec $\lambda_{concept}$ et $\lambda_{relation}$ comme paramètre de lissage

Sur cette collection, le modèle de GAO et le modèle que nous proposons fournissent des résultats similaires. Quelle que soit la méthode de calcul des estimations, nous ne trouvons aucune différence significative entre les deux. De même nous ne trouvons aucune différence significative entre ces deux modèles et celui basé uniquement sur les unigrammes. En précision moyenne, le modèle brut donne des résultats très légèrement supérieurs à ceux obtenus par la méthode aménagée, or cette méthode de pondération fournit un résultat binaire. Les relations sont précises, le simple fait qu'elles apparaissent ou non constitue une information suffisante pour discriminer les documents. Dans la suite nous n'utilisons que la pondération brute. Nous ne trouvons aucune différence significative entre le modèle à base de concepts et le modèle local, bien que l'utilisation de relations améliore la précision à 5 documents. Cependant, la correspondance des relations sans étiquette est proche d'un modèle de cooccurrence des termes au sein des phrases.

4 Modèle global avec étiquette

Nous testons dans cette partie le modèle global avec la prise en compte des étiquettes, nous évaluons ce modèle sur les trois méthodes d'extraction des concepts et nous comparons les résultats des ces variations du modèle global à celles obtenues par le modèle *MLC* en précision moyenne tableau 46 et en précision à 5 documents tableau 47. Nous vérifions que la prise en compte des étiquettes apporte un plus en recherche d'information.

méthode	$\lambda_{concept}$	entraînement	évaluation
MetaMap	0.1	**0.2725**	0.3371
MapMiniPar	0.1	0.2546	0.3390
MapTreeTagger	0.1	0.2663	**0.3653**

Tableau 46 Précision moyenne pour le modèle MLC appliqué aux trois méthodes de production de concepts, avec $\lambda_{concept}$ comme paramètre de lissage

méthode	$\lambda_{concept}$	entraînement	évaluation
MetaMap	0.2	0.4417	**0.4733**
MapMiniPar	0.1	0.4083	0.4533
MapTreeTagger	0.1	**0.4167**	0.4533

Tableau 47 Précision à 5 documents pour le modèle MLC appliqué aux trois méthodes de production de concepts, avec $\lambda_{concept}$ comme paramètre de lissage

D'une part, les trois méthodes montrent que l'apparition d'un concept dans le document fournit une information importante. Le lissage prend plus en compte la probabilité du concept dans le document (0.9) que dans la collection (0.1). D'autre part, les résultats du modèle *MLC* montrent un comportement différent des trois indexations sur la collection. Au niveau de l'apprentissage, MetaMap donne le meilleur résultat en précision moyenne alors que c'est MapTreeTagger qui donne les meilleurs résultats au niveau de l'évaluation où MetaMap donne le moins bon résultat avec une différence de 8%. La méthode MapMiniPar fournit globalement des résultats proches de la moins bonne des deux autres méthodes. Au niveau de la précision à 5 documents, MetaMap donne les meilleurs résultats alors que dans les approches précédentes MetaMap donne de meilleurs résultats en rappel plutôt qu'en précision.

4.1 Modèle avec probabilité simple

Le premier modèle de graphe évalué utilise une seule probabilité sur les étiquettes. Nous utilisons l'estimation brute pour calculer la probabilité d'une étiquette. Par conséquent, le système ajoute toutes les étiquettes possibles entre deux concepts définies dans le réseau sémantique d'UMLS, la probabilité de génération d'une relation devient binaire ; elle vaut 1 si la relation apparaît et $\lambda_{relation}$ sinon. Le tableau 48 présente les résultats pour la précision moyenne et le tableau 49 la précision à 5 documents.

méthode	$\lambda_{concept}$	$\lambda_{relation}$	entraînement	évaluation
MetaMap	0.1	0.4	**0.2806**	0.3437
MapMiniPar	0.1	0.4	0.2601	0.3486
MapTreeTagger	0.1	0.4	0.2707	**0.3722**

Tableau 48 Précision moyenne pour le modèle MG_{inter} appliqué aux trois méthodes de génération de concepts, avec $\lambda_{concept}$ et $\lambda_{relation}$ comme paramètre de lissage

méthode	$\lambda_{concept}$	$\lambda_{relation}$	entraînement	évaluation
MetaMap	0.1	0.7	**0.4583**	0.4600
MapMiniPar	0.1	0.1	0.4333	**0.4867**
MapTreeTagger	0.1	0.1	0.4333	0.4733

Tableau 49 Précision à 5 documents pour le modèle MG_{inter} appliqué aux trois méthodes de génération de concepts, avec $\lambda_{concept}$ et $\lambda_{relation}$ comme paramètre de lissage

Les résultats montrent une amélioration des performances par l'utilisation du modèle MG_{inter}, quelle que soit la méthode d'extraction des concepts utilisée. Cette amélioration reste cependant faible, elle se situe aux environs de 2% pour la méthode MapTreeTagger. Nous remarquons aussi que le meilleur $\lambda_{relation}$ pour la précision moyenne est le même pour les trois méthodes proposées. Le modèle a donc le même comportement sur ces méthodes de génération des concepts. Au niveau de la précision à 5 documents, la méthode MapMiniPar donne la meilleure précision. Par ailleurs, nous remarquons que la meilleure précision sur les détections de concepts MapMiniPar et MapTreeTagger s'obtient pour un

$\lambda_{relation}$ de 0.1, c'est-à-dire donnant de la force au document. MetaMap, pour sa part, utilise un paramètre de 0.7 pour lequel l'amélioration de la précision se révèle moins forte. Les relations obtenues avec MetaMap semblent moins précises, par conséquent cette méthode n'améliore par la précision.

4.2 Modèle avec probabilité décomposée

Nous testons ici la décomposition du calcul de la probabilité d'une relation étiquetée $P\left(uv'_{relation} \middle| uv'_{concept1}, uv'_{concept2}, M_D^g\right)$. Cette probabilité se décompose en deux contributions, l'une correspond à la probabilité de générer le couple qui forme la relation, et l'autre à la probabilité d'attribuer à ce couple l'étiquette désirée. Cette méthode permet d'avoir deux facteurs qui prennent en compte des informations différentes. Pour l'estimation de la contribution du couple $P\left(uv'_{couple} \middle| uv'_{concept1}, uv'_{concept2}, M_D^g\right)$, nous utilisons le calcul brut, c'est à dire le nombre d'apparitions du couple dans un graphe de phrases divisé par le nombre d'apparitions de ces concepts dans un graphe de phrases. Cette estimation donne 1 si le couple apparaît dans le document et λ_{couple} sinon. Pour l'estimation de la génération d'une étiquette associée à un couple $P\left(uv'_{relation} \middle| uv'_{couple}, M_D^g\right)$, nous n'utilisons pas directement la probabilité brute, car celle-ci fournit une constante pour chaque couple de concepts et ne donne pas de bons résultats en recherche d'information. Nous proposons d'utiliser une estimation plus globale de l'attribution d'une étiquette : le nombre de relations étiquetées par cette étiquette dans le document divisé par le nombre de relations étiquetées sur le document. Comme pour les modèles précédents nous évaluons la précision moyenne (tableau 50) et la précision à 5 documents (tableau 51).

méthode	$\lambda_{concept}$	λ_{couple}	$\lambda_{relation}$	entraînement	évaluation
MetaMap	0,1	0,4	0,9	0.2650	0.3244
MapMiniPar	0,1	0,5	0,8	0.2605	0.3466
MapTreeTagger	0,1	0,3	0,8	0.2698	**0.3673**

Tableau 50 Précision moyenne pour le modèle avec probabilité décomposée appliqué aux trois méthodes de génération de concepts avec $\lambda_{concept}$, λ_{couple} et $\lambda_{relation}$ comme paramètre de lissage

méthode	$\lambda_{concept}$	λ_{couple}	$\lambda_{relation}$	entraînement	évaluation
MetaMap	0,1	0,7	0,5	0.4083	0.4200
MapMiniPar	0.1	0.1	0.9	0.4083	**0.4867**
MapTreeTagger	0,1	0,1	0,9	0.4167	**0.4867**

Tableau 51 Précision à 5 documents pour le modèle avec probabilité décomposée appliqué aux trois méthodes de génération de concepts avec $\lambda_{concept}$, λ_{couple} et $\lambda_{relation}$ comme paramètre de lissage

Par l'utilisation de cette méthode, les résultats contrastent. Les résultats obtenus restent globalement inférieurs aux résultats du modèle sans décomposition des probabilités. Les performances de la méthode MetaMap diminuent fortement, et bien que les résultats des deux autres méthodes le soient aussi, elles diminuent beaucoup moins. La précision de MetaMap se dégrade aussi fortement. Nous remarquons que le $\lambda_{relation}$ qui donne les meilleurs résultats devient 0.9, la contribution de la collection devient beaucoup plus forte que celle du document. Cela provient essentiellement du fait que le calcul de cette probabilité donne des probabilités fortement différentes entre la collection et le document, cette probabilité obtenant un résultat beaucoup plus fort sur les documents que sur la collection.

Ce modèle permet de prendre en compte les étiquettes de manière plus complète en utilisant deux contributions. Cependant ce modèle ne fonctionne pas complètement car il ne désambiguïse pas les étiquettes. Il peut cependant servir de base à l'intégration d'un score d'incertitude sur les relations, en intégrant ce score lors du calcul des probabilités d'attribution des étiquettes. Par conséquent, ce modèle n'est pas inintéressant pour de futurs travaux.

4.3 Synthèse

Nous proposons deux processus génératifs qui prennent en compte les étiquettes des relations dans la modélisation d'un graphe de concepts. Ces modèles s'intègrent dans notre modèle de recherche d'information *MG*. Le modèle sans décomposition ressort comme le modèle le plus efficace. Ce modèle se heurte cependant au fait que les relations telles que notre méthode les extraits ne permettent pas de prendre en compte toute la portée des étiquettes. Afin de mieux appréhender ces étiquettes, nous proposons de décomposer la probabilité de génération d'une relation étiquetée en deux facteurs, mais cette décomposition n'aboutit pas en une amélioration des résultats. Les méthodes de génération de relations n'effectuent pour l'instant aucune désambiguïsation des étiquettes, et ces modèles ne prennent pas en compte les scores de confiance. Nous pensons néanmoins qu'il serait intéressant d'approfondir le deuxième modèle afin de prendre en compte les scores de confiance sur les étiquettes au sein des modèles de graphes, sachant que ces scores ont montré un intérêt pour améliorer la précision dans le modèle précédent. Dans la suite nous utilisons le modèle sans décomposition des relations.

5 Expérimentations complémentaires

5.1 Type de requête

Comme nous l'avons vu dans la présentation de la collection CLEF médicale, les requêtes de CLEF 2006 et 2007 se classent selon trois types : les requêtes visuelles, textuelles et mixes. Nous évaluons dans cette partie l'apport de notre modèle en fonction du type de requête utilisé. Pour cela, nous évaluons notre modèle sur les diagnostics en anglais constitués du regroupement de la collection CLEF 2007 et CLEF 2006 : EN_DIAG_ 0607. Sur cette évaluation nous découpons les requêtes en trois ensembles en fonction de leur type, et nous testons les modèles *MLC* et *MG* avec les paramètres obtenus dans la partie précédente sur ces ensembles. Nous donnons dans le tableau 52 les résultats en précision moyenne et en précision à 5 documents pour la méthode MapTreeTagger et dans le tableau 53 les résultats équivalents pour la méthode MetaMap.

	MAP		P@5	
Aspects	unigramme	MG	unigramme	MG
Mixte	0.3145	**0.3239**	0.4300	0.4300
Visuel	0.2360	0.2362	0.4105	**0.4211**
Texte	0.3861	0.3857	0.7000	0.7000

Tableau 52 Précision moyenne et précision à 5 documents sur EN_DIAG_0607 avec MapTreeTagger en fonction des aspects des requêtes sur le modèle global

	MAP		P@5	
Aspects	unigramme	MG	unigramme	MG
Mixte	0.2990	**0.3269**	0.3800	**0.4600**
Visuel	**0.2002**	0.1915	**0.3700**	0.3600
Texte	0.3886	0.3910	0.6700	0.6700

Tableau 53 Précision moyenne et précision à 5 documents sur EN_DIAG_0607 avec MetaMap en fonction des aspects des requêtes sur le modèle global

Les résultats montrent que le modèle MG apporte essentiellement des améliorations sur les requêtes de type mixte. Avec la méthode MapTreeTagger les résultats sur les autres types de requêtes restent stables, même s'ils diminuent légèrement. Avec MetaMap les résultats sur les requêtes images se dégradent fortement. L'utilisation des concepts est suffisante pour les requêtes textuelles, cela se reflète par la force de leur précision à 5 documents. L'apport des relations améliore les requêtes mixtes car ces requêtes mettent plus en avant le caractère relationnel des informations recherchées, ce sont des requêtes expertes. Au contraire, les requêtes textuelles et visuelles peuvent dans certains cas être constituées d'un simple concept (cf. tableau 54), ce sont des requêtes plus simples. Prendre en compte le type de requête permettrait d'obtenir des résultats plus adaptés aux besoins des utilisateurs.

	requête	
Type	Année et numéro	texte
Visuel	2007 n°4	Radio d'une fracture de la hanche
Textuel	2007 N°25	Sclérose tubaire
	2007 N°24	Carcinome gastrointestinal
Mixte	2007 N°12	Endoscopie grastroentestinale avec polype
	2007 N°16	CT d'un abcès du foie

Tableau 54 Exemple de requêtes en fonction de leur type.

6 Multilingue et multi-Extraction (CLEF 2007)

Comme nous l'avons vu, les trois méthodes d'extraction des concepts possèdent chacune leurs caractéristiques. Dans cette partie nous étudions le regroupement des différentes méthodes, notamment au niveau des requêtes. Nous analysons la partie anglaise de la collection à l'aide de MetaMap et les parties française et allemande à l'aide de la méthode MapTreeTagger.

Pour les requêtes nous proposons de regrouper les différentes analyses. Une requête est alors constituée d'un ensemble de graphes, $q = \{G_Q\}$. La probabilité d'une requête q sachant un modèle de document s'obtient alors par la multiplication des probabilités de génération de chaque graphe de l'ensemble.

$$P\left(q = \{G_Q\} \middle| M_D^g\right) = \prod_{G_Q} P\left(G_Q \middle| M_D^g\right)$$

Nous proposons les regroupements de requêtes suivants :

- **E** : le graphe de la requête anglaise construit avec la méthode MetaMap.
- **E_Mix** : l'ensemble formé des trois graphes détectés sur le texte anglais à l'aide des trois méthodes de détection des concepts.

- **EFG** : l'ensemble formé de trois graphes constitués de l'analyse par MetaMap sur la requête anglaise et de celle de MapTreeTagger sur le français et l'allemand.
- **EFG_Mix** : l'ensemble formé de cinq graphes constitués des graphes obtenus par MapTreeTagger sur les trois langues associées au graphe obtenu par MapMiniPar et MetaMap sur l'anglais.

Sur ces différentes requêtes, nous étudions la variation des résultats lors de l'utilisation de plusieurs graphes, que ce soit avec les concepts seulement, méthode *MLC* (tableau 55) ou pour le modèle global (tableau 56), nous effectuons pour cela un apprentissage sur la partie textuelle des collections CLEF 2005 et 2006 et nous évaluons les résultats au niveau image de CLEF 2007. Nous donnons par ailleurs les résultats pour les collections 2005 et 2006 au niveau image.

	$\lambda_{concept}$	A_TXT_0506	A_IMG_0506	A_IMG_07
MAP				
E	0.2	0.2468	0.2284	0.3131
E_mix	0.1	0.2610	0.2359	0.3376
EFG	0.1	0.2547	0.2274	0.3269
EFG_mix	0.1	0.2673	0.2395	**0.3538**
P@5				
E	0.2	0.4618	0.4436	0.3733
E mix	0.1	0.4727	0.4582	0.3667
EFG	0.2	0.4582	0.4364	**0.4467**
EFG_mix	0.1	0.4836	0.4691	0.4200

Tableau 55 Précision moyenne et précision à 5 documents pour le regroupement de requêtes sur les concepts

	$\lambda_{concept}$	$\lambda_{relation}$	A_TXT_0506	A_IMG_0506	A_IMG_07
MAP					
E	0.2	0.9	0.2463	0.2277	0.3271
E_mix	0.1	0.9	0.2620	0.2363	0.3377
EFG	0.1	0.9	0.2556	0.2313	0.3345
EFG_mix	0.1	0.9	0.2670	0.2394	**0.3536**
P@5					
E	0.2	0.9	0.4582	0.4400	0.4133
E_mix	0.1	0.8	0.4800	0.4582	0.3667
EFG	0.1	0.8	0.4618	0.4473	**0.4867**
EFG_mix	0.1	0.8	0.4909	0.4655	0.4200

Tableau 56 Précision moyenne et précision à 5 documents pour le regroupement de requêtes sur le modèle global

Les résultats montrent que la méthode la plus performante pour la précision moyenne est la méthode qui utilise toutes les sources de concepts présentées dans cette section (EFG mix). Multiplier les sources de détections de concepts sur la requête permet d'améliorer les résultats globaux de recherche d'information. Une telle méthode permet de retrouver tous les concepts exprimés dans les requêtes et par conséquent améliore le rappel. Par contre, pour la précision à 5 documents, les meilleurs résultats s'obtiennent avec la méthode EFG qui n'utilise qu'une source de concepts par langue. Utiliser seulement les concepts détectés sur les trois langues détecte les concepts les plus pertinents pour la tâche de recherche, ajouter d'autres sources de concepts augmente le nombre de concepts moins précis et par conséquent diminue la précision.

A l'indexation, le modèle *MG* fournit le même comportement que les concepts seuls. L'unique cas où l'utilisation des relations améliore les résultats du modèle *MLC* est le regroupement des trois langues EFG. Dans ce cas, les résultats du modèle *MG* améliorent la précision moyenne et la précision à 5 documents. Cela confirme que cette méthode extrait mieux les concepts et les relations de la requête que les autres. Par conséquent, cette méthode fournit les meilleurs résultats en P5D lors de l'utilisation des relations.

Cette méthode nous a permis d'obtenir les meilleurs résultats lors de la campagne d'évaluation CLEF médicale de 2007[26].

7 Conclusion

Cette partie démontre que l'utilisation du modèle global améliore les résultats du modèle de langue conceptuel *MLC*. Le modèle global sans étiquette fournit des résultats similaires à ceux obtenus par les concepts, mais aussi similaires à ceux obtenus par le modèle proposé par GAO. Nous testons ensuite le modèle global avec la prise en compte des étiquettes et nous évaluons les différentes variations de ce modèle proposé dans cette thèse. Ces résultats montrent à nouveau que l'utilisation d'un multi-index composé d'une vision relationnelle et d'une vision conceptuelle du document améliore la précision des résultats. Sur les différents modèles proposés, la méthode de génération de graphes qui considère l'intersection des étiquettes et ne décompose pas la probabilité de relations étiquetées donne donc les meilleurs résultats.

Nous testons ensuite, à l'aide de ce modèle, les types de requêtes. Les résultats montrent que ce modèle améliore les requêtes mixes, c'est-à-dire celles qui mettent en avant les aspects visuel et textuel ; cela donne un résultat intéressant car ce type de requêtes correspond à des besoins experts. C'est ce type de besoins que l'on souhaite résoudre dans les systèmes orientés précision. Nous montrons enfin à l'aide d'une combinaison des requêtes que nos différentes méthodes d'extraction des concepts peuvent se combiner notamment au niveau des requêtes. Cette combinaison améliore les résultats, et nous a permis d'obtenir les meilleures performances dans une évaluation inter-laboratoire. Nous comparons par la suite ce modèle avec le modèle local.

[26] http://ir.ohsu.edu/image/2007results.html

Conclusion

Les deux chapitres précédents évaluent les deux modèles orientés précision proposés. Dans cette partie nous comparons les résultats de ces deux modèles et nous concluons sur ces expérimentations.

1 Comparaison des modèles

Nous comparons ici les meilleurs résultats obtenus pour chacun des deux modèles sur les trois méthodes d'extraction des concepts. Nous utilisons pour cela la partie évaluation de la collection proposée pour le modèle global et nous ne présentons que les résultats obtenus au niveau des diagnostics.

méthode	concept	relation	évaluation MAP	évaluation P@5
MapMiniPar	Tf	Tf + confiance	0.2893	**0.4933**
	Tf	Idf + confiance	0.2875	**0.4800**
MetaMap	Idf + confiance	**aucune**	0.2802	0.3733
MapTreeTagger	Tf	Tf.Idf	**0.2935**	0.4333
	occ	occ	0.2929	0.4400

Tableau 57 Résultats en précision moyenne et en précision à 5 documents sur la partie évaluation de EN_DIAG_ 0506 pour les meilleurs résultats du modèle local

méthode	$\lambda_{concept}$	$\lambda_{relation}$	évaluation
MAP			
MetaMap	0.1	0.4	0.3437
MapMiniPar	0.1	0.4	0.3486
MapTreeTagger	0.1	0.4	**0.3722**
P@5			
MetaMap	0.1	0.7	0.4600
MapMiniPar	0.1	0.1	**0.4867**
MapTreeTagger	0.1	0.1	0.4733

Tableau 58 Résultats en précision moyenne et en précision à 5 documents sur la partie évaluation de EN_DIAG_ 0506 pour les meilleurs résultats du modèle global

Ces résultats montrent qu'en précision moyenne le modèle global est largement supérieur au modèle local avec 26 % de différence. Nous remarquons que le modèle global nécessite une étape d'apprentissage pour déterminer les bons paramètres à utiliser.

Au niveau de la précision à 5 documents les résultats sont plus partagés. Le modèle local donne le meilleur résultat, mais ce dernier reste proche des résultats obtenus par le modèle global, or le modèle global n'utilise pas pour l'instant les informations telles que la confiance sur les relations.

Au niveau de l'utilisation du support de vocabulaires, utiliser une représentation des documents qui porte sur tout le vocabulaire permet d'obtenir de meilleurs résultats en précision moyenne. Ces modèles intègrent mieux le contexte des documents que les modèles qui n'utilisent que le contenu des documents. Cependant utiliser des représentations sur la globalité du vocabulaire est souvent plus complexe, cela nécessite par exemple un apprentissage. Sachant que les deux modèles ont des précisions à 5 documents assez proches, utiliser des représentations sur tout le vocabulaire ne semble pas tellement pertinent dans des systèmes qui s'intéressent plus à la précision, du fait de leur complexité. Au niveau de la taille du vocabulaire, ces travaux ne permettent pas de donner de réponses précises, en effet comme nous l'avons utilisé le modèle local n'exploite pas cette possibilité. La méthode utilisée prend en compte les éléments qui n'appartiennent pas à la collection et qui sont détectés dans les requêtes, cependant ceux-ci n'impactent pas le résultat.

2 Bilan

2.1 Bilan sur les modèles

Dans ce chapitre, nous avons testé les différentes méthodes d'extraction des concepts et des relations. Les concepts fournissent des performances égales voire supérieures aux résultats des lemmes. Ces performances montrent que l'utilisation d'une représentation de niveau sémantique améliore les résultats obtenus à l'aide d'un modèle à base de lemmes, notamment au niveau de la précision. Nous testons ensuite les deux modèles orientés précision proposés. Sur ces deux modèles, les résultats montrent que l'utilisation d'un support de vocabulaires multi-index, avec un point de vue conceptuel et un point de vue relationnel, améliore la précision obtenue par les concepts seuls, et cela pour les deux modèles. Cela confirme le fait que les modèles possédant la plus forte expressivité fournissent les meilleurs résultats pour des requêtes expertes.

Nous proposons trois méthodes pour détecter les concepts. Au niveau de ces méthodes, les conclusions restent toujours les mêmes quel que soit le modèle. La méthode MapTreeTagger donne les meilleurs résultats en précision moyenne et la méthode MapMiniPar les meilleurs résultats en précision à 5 documents. Dans l'ensemble le modèle global fournit les meilleurs résultats, néanmoins des travaux restent à accomplir sur cette méthode, notamment pour intégrer le score de confiance sur les concepts et sur les relations, lequel a montré qu'il pouvait améliorer la précision des résultats.

A travers ces expérimentations, nous explorons différentes variations et possibilités d'utilisation des modèles à base de graphes. Nous testons différentes variations du *tf.idf* sur le modèle local et plusieurs variations du modèle de graphe sur le modèle global.

Au niveau du modèle local, les résultats montrent que le *tf* seul suffit si la méthode détecte des concepts peu nombreux et assez *précis*. Au contraire l'*idf* devient intéressant lorsque le nombre de concepts détectés devient grand et que les concepts sont moins précis comme avec MetaMap. Le modèle local, qui permet l'intégration du score de confiance, montre l'intérêt de ce score notamment sur les relations.

Au niveau du modèle global, les résultats montrent que les modèles de graphe qui considèrent les étiquettes entre concepts comme complémentaires donnent de meilleurs résultats que les modèles considérant ces étiquettes comme indépendantes. De plus, si la décomposition de la probabilité des relations étiquetées n'améliore pas les résultats, elle reste une voie envisageable pour l'intégration de nouvelles connaissances dans le modèle global.

En comparant les résultats des deux modèles, le modèle global fournit dans l'ensemble de meilleurs résultats que le modèle local. Les deux modèles utilisent des approches différentes de l'utilisation du support de vocabulaires pour représenter les documents. Suite à ces expérimentations, nous pouvons

dire que le modèle global s'adapte bien à l'utilisation de corpus de taille fixe ou 'connue' dont on connaît les propriétés, d'une part car la mise à jour du modèle lors de l'ajout de nouveaux documents est complexe puisqu'il utilise tous les éléments du support de vocabulaires, d'autre part car ce modèle nécessite un apprentissage, bien qu'ici les paramètres restent assez stables. Le modèle local quant à lui s'adapte à des corpus de tailles variables évoluant au cours du temps, par exemple des applications web, les performances de ce système fournissent en effet de bons résultats en précision à 5 documents. La représentation d'un document dans ce système peut facilement se modifier, notamment avec l'utilisation de pondérations en *tf* seuls qui donnent de bons résultats.

Au final nous avons exploré l'utilisation de modèles de recherche d'information orientés précision, par l'utilisation de deux modèles expressifs. L'expressivité de ces deux modèles, exprimés par le support de vocabulaires, permet d'obtenir une amélioration de la précision. Cette expressivité passe par l'utilisation d'un point de vue conceptuel et d'un point de vue relationnel. Elle passe de plus par l'utilisation d'informations supplémentaires telles que le score de confiance. Pour un même niveau d'expressivité, nos deux modèles montrent que les différentes méthodes d'utilisation des vocabulaires donnent des moyens pour adapter les modèles au contexte d'utilisation.

2.2 Bilan sur les campagnes d'évaluations

Les expérimentations présentées dans ce chapitre nous ont permis de développer un système efficace pour résoudre les requêtes de la campagne d'évaluation CLEF médicale. Les résultats obtenus sur cette campagne découlent aussi de résultats obtenus lors de notre participation à d'autres campagnes d'évaluations, internationales ou nationales.

2.2.1 Clef médicale

Le modèle global nous a permis d'obtenir les meilleurs résultats lors de la campagne d'évaluation CLEF médicale alors que nous n'avons utilisé que la partie textuelle de cette collection. Ces résultats valident plusieurs aspects de nos travaux : l'utilisation des concepts et les méthodes d'extraction retenues, l'utilisation des modèles de langue sur les concepts et leur extension aux graphes, et enfin les complémentarités des méthodes d'extraction.

2.2.2 Autres campagnes

Nous avons participé à d'autres campagnes d'évaluation et ces dernières ont impacté sur un certain nombre de nos choix :

- En 2005 nous avons participé à la campagne d'évaluation DEFT05 dont le but était l'attribution automatique d'un discours à son auteur. Cette première campagne a permis d'établir l'utilité d'approches combinant des lemmes et des relations syntaxiques sur certains domaines. Sur cette campagne, nous avons obtenu la huitième position sur 25 participations.
- Par ailleurs, la même année nous avons participé à la campagne CLEF05 où nous avons utilisé les lemmes et les relations syntaxiques pour une tâche de recherche d'information. Nous avons également testé l'utilisation de modèles de langue sur la structure syntaxique. Cette campagne nous a permis de mettre en évidence le besoin d'une expressivité de niveau sémantique. Nos résultats sur cette campagne se sont situés dans la moyenne des participants.
- En 2006, nous avons participé à la campagne d'évaluation DEFT06 dont le but était la segmentation thématique de documents. Pour cette participation, nous avons montré l'intérêt de la combinaison de différentes approches pour la segmentation. Cette approche a permis à notre équipe d'obtenir le deuxième meilleur résultat.

PARTIE 6 : CONCLUSION

Bilan

« Dans les sciences, le chemin est plus important que le but. Les sciences n'ont pas de fin. » Erwin Chargaff (A la découverte de la science)

Cette thèse propose un cadre général qui permet de développer des modèles orientés précision. Ce cadre met en avant la notion de supports de vocabulaires qui définit le modèle de la requête et le modèle de document utilisé par le système. Ces supports de vocabulaires modélisent l'expressivité des représentations. Ils la modélisent d'une part à l'aide de plusieurs points de vue ; d'autre part à l'aide de points de vue expressifs exprimés par des vocabulaires complexes et pondérés.

Actuellement, l'expressivité ne figure pas de manière formelle dans les systèmes. Cela constitue un inconvénient quand on souhaite évaluer ou manipuler cette expressivité. Ce cadre de modélisation basé sur des supports de vocabulaires est une contribution majeure. En effet, les supports de vocabulaires mettent en avant l'expressivité des modèles, cela permet de positionner les modèles les uns par rapport aux autres selon leur expressivité. Cela permet aussi d'étudier et d'adapter le niveau d'expressivité à la tâche effectuée.

Dans ce cadre nous nous sommes orientés vers l'utilisation d'une représentation expressive du texte. Nous avons proposé deux modèles utilisant des représentations d'expressivité forte : le modèle local et le modèle global. Ces deux modèles se basent sur un même support de types formé de types sémantiques. Ils définissent des supports de vocabulaires offrant une vision conceptuelle et une vision relationnelle du document à l'aide de vocabulaires complexes. Si ces deux modèles se ressemblent au niveau de l'expressivité, ils s'opposent sur l'utilisation de leur support de vocabulaires. Le modèle local utilise des vocabulaires exhaustifs et il ne représente ensuite un document qu'avec une partie de ses vocabulaires, celle qui correspond au document. Le modèle global pour sa part limite ses vocabulaires aux unités spécifiques de la collection et représente un document sur l'ensemble de ses vocabulaires.

À partir de ces choix, nous avons implémenté le modèle local à l'aide d'un modèle dérivé des graphes conceptuels et le modèle global par un modèle original dérivé des modèles de langue.

Sur du texte, l'obtention de ces modèles repose sur la génération d'une représentation intermédiaire constituée d'une représentation pour chaque phrase du document. La représentation finale du document constitue alors une synthèse des représentations de phrase du document, cette synthèse dépend du modèle et met en avant le thème du document.

La représentation de phrase s'obtient à l'aide d'outils de traitement de la langue. Dans le but de prendre en compte la difficulté de ces traitements, nous proposons d'intégrer à cette représentation des scores de confiance. Ces scores représentent la confiance dans la détection des éléments qui constituent les représentations intermédiaires. Nous nous servons ensuite de cette représentation intermédiaire pour construire les deux modèles. L'avantage de cette représentation intermédiaire est qu'elle permet de mutualiser certains traitements sur plusieurs domaines : appliquer les deux modèles à un domaine ne requiert que la construction de la représentation intermédiaire des documents. L'utilisation de nos modèles est donc simplifiée et cette approche facilite leur portabilité à différents domaines.

La construction de la représentation intermédiaire sur du texte général est difficile. C'est une représentation de niveau sémantique qui nécessite de résoudre des problèmes linguistiques. Pour faciliter sa construction, nous proposons de construire cette représentation sur un domaine précis, le domaine médical. Nous utilisons pour cela le thésaurus UMLS qui permet de définir les types sémantiques qui forment le support de types et nous proposons plusieurs méthodes pour construire la représentation intermédiaire des documents.

Nous évaluons finalement l'application des deux modèles sur le domaine médical à travers la collection CLEF médicale. Cette collection nous permet de tester notre modèle dans un cadre réel et de l'évaluer vis-à-vis d'autres équipes. Nous évaluons dans un premier temps les différentes méthodes de création de la représentation intermédiaire, puis nous testons et comparons les deux modèles.

Les résultats basés sur l'utilisation des supports de vocabulaires montrent l'intérêt d'utiliser des représentations d'expressivité forte. Ces représentations sont obtenues par des traitements de la langue fournissant une forte couverture du contenu et par l'utilisation de score de confiance. Sur les deux modèles, l'utilisation de vocabulaires expressifs, concepts et relations sémantiques, améliore les résultats. De plus, sur le modèle local, l'utilisation d'un score de confiance améliore la précision des réponses. Ces méthodes prouvent l'intérêt des indexations conceptuelles et montrent qu'il est possible de les automatiser sur un domaine précis.

La construction de la représentation des phrases étant une tâche multilingue et difficile, nous avons utilisé différentes méthodes de détection des concepts. Ces méthodes fournissent des résultats différents, certaines fournissent de bonnes précisions, d'autres de meilleurs rappels. Une dernière expérimentation montre la complémentarité de ces détections et le fait qu'elles apportent chacune des informations importantes. Cette complémentarité souligne bien, en recherche d'information, l'importance de couvrir le contenu du document, quitte à extraire des informations imprécises.

Nous proposons dans cette thèse deux modélisations sémantiques basées sur deux approches différentes de la portée des vocabulaires et des représentations. L'efficacité de ces deux modèles est prouvée par un certain nombre d'expérimentations sur la collection de la campagne d'évaluation CLEF médicale et par la participation à cette campagne en 2007 où le modèle global nous a permis d'obtenir la première place.

Nos expérimentations montrent l'importance de l'axe de la portée des représentations, cependant l'augmentation de la portée de la représentation induit une complexité plus forte du modèle. Il est alors nécessaire de trouver un juste milieu entre les performances et la complexité du modèle, notamment dans les domaines où les collections varient rapidement.

L'axe de la portée du vocabulaire a quant à lui un impact plus faible sur les résultats il influe surtout sur la capacité du modèle à prendre en compte des informations nouvelles.

Cette thèse prouve que l'utilisation de modèles possédant une forte expressivité constitue un apport en recherche d'information orientée précision. Ces modèles permettent de prendre en compte le maximum de spécificités du document. Ils permettent de représenter plusieurs points de vue sur le document, d'exprimer des informations complexes et de niveau sémantique. Cela est démontré par nos résultats à la campagne d'évaluation CLEF médicale.

Cette thèse propose une méthode innovante pour développer des modèles, qui se base sur l'utilisation de supports de vocabulaires. Ces supports de vocabulaires mettent en avant des propriétés qui permettent, à des niveaux d'expressivité semblables, de construire des modèles en fonction des caractéristiques de l'application. L'utilisation de ces supports nécessite donc de faire des choix sur l'utilisation du *support de vocabulaires de documents*, cela en fonction des caractéristiques de la tâche à effectuer.

Perspectives

1 Court terme

Sur le court terme, les améliorations peuvent se présenter sous deux axes, premièrement l'amélioration des méthodes de détection de graphes, et deuxièmement l'amélioration des modèles.

1.1 Amélioration de la représentation intermédiaire

Nous proposons trois méthodes de détection des concepts. Sur les méthodes à base d'analyseurs morphosyntaxiques, nous pourrions utiliser un analyseur adapté au domaine médical. Ce serait d'autant plus intéressant si cet analyseur se révélait capable de prendre en compte les particularités des diagnostics qui utilisent un style de texte *télégraphié*. De tels analyseurs morphosyntaxiques existent pour ce type de textes, nous devons cependant choisir des analyseurs assez robustes et rapides. Nous souhaiterions utiliser des analyseurs syntaxiques qui fournissent une analyse en dépendance pour nous permettre de calculer des scores de confiance sur les relations, voire sur les concepts.

Nous pouvons aussi nous intéresser à l'interaction des concepts et des termes au sein de la phrase pour éliminer certains concepts ou pour leur attribuer un score de confiance. Nous montrons que nos trois méthodes d'extraction des concepts sont complémentaires, cependant leur complémentarité n'est testée que sur les requêtes. Nous pourrions utiliser leur complémentarité sur les documents, soit dans le but de calculer un score de confiance sur les concepts en s'inspirant des méthodes de combinaison d'analyseurs syntaxiques proposées par (Brunet-Manquat, 2004), soit dans le modèle global pour mieux modéliser les documents.

Sur les relations, nous utilisons une méthode qui ne prend en compte que des informations de cooccurrence au sein des phrases. Utiliser d'autres informations permettrait de mieux détecter les relations. Nous pourrions par exemple détecter des relations entre les phrases en utilisant des méthodes proches de (Vechtomova *et al.*, 2006). Cette méthode utilise les contextes au sein des phrases afin d'élargir la portée des relations. Nous pourrions aussi utiliser des informations de cooccurrence entre les relations pour tenter de sélectionner certaines étiquettes.

Nous pourrions améliorer notre utilisation des informations syntaxiques pour le calcul du score de confiance. Certains travaux se basent sur des indices provenant des verbes utilisés dans les phrases. Ne pas considérer tous les éléments d'un chemin syntaxique comme équivalents, et donner plus de force à certains, tels que les verbes, améliorerait nos résultats. Globalement, l'utilisation d'un corpus d'apprentissage pourrait s'avérer bénéfique pour améliorer nos méthodes de détection et pour calculer des scores de confiance. De tels corpus restent peu disponibles, nous pouvons cependant nous interroger sur les méthodes semi-supervisées pour améliorer les méthodes de détection.

Nous envisageons d'appliquer nos modèles et nos méthodes sur d'autres corpus médicaux que le corpus CLEF médicale, cela permettrait de valider nos méthodes de détection et de vérifier la portabilité de ces méthodes. Pour cela nous nous intéressons notamment à la tâche biomédicale de TREC qui depuis 2007 se rapproche d'une tâche de recherche d'information.

1.2 Parfaire les modèles

Les deux modèles que nous proposons ne prennent pas en compte les relations paradigmatiques, développer l'un ou les deux modèles pour prendre en compte ces relations constitue une perspective intéressante. Globalement l'apport d'autres informations ou d'informations complémentaires aux relations et aux concepts doit s'envisager dans nos modèles, plus précisément sur le modèle global. En effet, ce modèle ne prend pas en compte le score de confiance, or d'après nos expérimentations, ce score paraît intéressant en recherche d'information. À court terme nous souhaitons développer le modèle global pour intégrer cette pondération, soit directement au niveau du modèle, soit lors du calcul des estimations du modèle.

2 Long terme

Ce travail ouvre plusieurs voies, d'une part en appliquant les deux modèles présentés à de nouvelles données, d'autre part en explorant les possibilités mises en avant par l'utilisation de support de vocabulaires.

2.1 Élargir les applications

Premièrement nous envisageons d'appliquer les deux modèles sur d'autres domaines que le domaine médical. Pour cela nous devons sélectionner des domaines d'application où la construction de graphes est possible, c'est-à-dire un domaine disposant des ressources nécessaires. Pour chacun des domaines où nous souhaitons appliquer nos modèles, nous devons définir le support de types utilisé et les méthodes permettant d'extraire les concepts et les relations sur les phrases. Nous montrons en annexe que ces modèles s'appliquent sur des représentations syntaxiques de surface. Nous pourrions tester l'utilisation de ces modèles sur des représentations syntaxiques plus profondes, par exemple UNL[27], cela pourrait constituer une première étape vers un portage sur d'autres domaines.

En plus de l'élargissement des modèles à d'autres domaines, nous pourrions aussi nous intéresser à leur application sur d'autres médias, ou même à leur application sur des données multimédias. Cela serait par exemple possible sur la collection CLEF médicale où les modèles pourraient prendre en compte des concepts détectés sur les images. Nous pourrions alors nous poser la question de l'intérêt de l'ajout de points de vue spécifiques aux images et de l'utilisation des vocabulaires existant pour représenter les images, par exemple la détection de certains concepts d'UMLS dans les images. Par conséquent nous devrions aussi nous poser la question de l'intégration de ces vocabulaires dans les fonctions de correspondance.

L'utilisation de graphes permet d'améliorer les résultats des requêtes, cependant cela nécessite d'avoir des requêtes précises contenant des relations. Sur Internet, les utilisateurs posent rarement de telles requêtes. Les utilisateurs sont habitués à utiliser des systèmes peu expressifs et peuvent avoir des difficultés pour exprimer de telles requêtes. L'expression d'une requête textuelle reste plus restrictive que l'expression d'un sens. Nous pourrions envisager un modèle de recherche d'information qui permette à l'utilisateur d'exprimer ses besoins sous forme sémantique, soit en l'aidant à sélectionner les concepts et les relations exprimés par sa requête textuelle, soit en l'aidant à créer directement un graphe conceptuel comme requête. Cela nécessiterait de bien comprendre les utilisateurs qui expriment des besoins experts et de pourvoir tester les méthodes de création des requêtes.

[27] http://www.undl.org/

2.2 Explorer les dimensions dues à l'utilisation de supports de vocabulaires

Nous avons développé deux modèles qui explorent les dimensions que nous nommons par utilisation du support et construction du support. L'utilisation du support de vocabulaires pour créer les modèles de documents et de requêtes permet l'exploration de ces dimensions. Nous avons cependant exploré deux modélisations assez distinctes. Nous pourrions par la suite proposer d'autres modèles qui ne varient les uns par rapport aux autres que sur une seule de ces dimensions. Cela nous permettrait de juger plus précisément de la pertinence de ces dimensions. Sur ces deux axes nous avons proposé des modèles qui utilisent tous leurs vocabulaires de la même manière, nous pourrions explorer des modèles dans lesquels tous les vocabulaires ne seraient pas exploités de la même façon. Par exemple nous pourrions utiliser une représentation qui utilise toutes les unités du vocabulaire de concept, mais seulement les relations détectées dans le document.

Ces supports ont l'avantage de nous permettre d'explorer d'autres choix sur leur utilisation. Nous pensons que nous pourrions dans la suite de nos travaux proposer des expérimentations qui évaluent différents types de positionnement par rapport au support de vocabulaires.

ANNEXES

Annexe A. Validité des Vocabulaires

Les graphes dont nous nous servons utilisent des vocabulaires spécifiques, aussi faut-il s'interroger sur l'utilisation de ces vocabulaires. La recherche d'information classique sur le texte est basée sur plusieurs hypothèses concernant la distribution des termes dans les documents.

1 Loi de Zipf

La loi de Zipf est une loi empirique énoncée en 1949 par G.K Zipf (Zipf, 1949), qui décrit la répartition statistique des fréquences d'apparition des différents éléments d'un ensemble, comme les mots d'un texte. Selon Zipf, les symboles d'un ensemble organisé typologiquement ne s'organisent pas de manière aléatoire mais suivant une loi de puissance. Si on classe les mots dans l'ordre décroissant de leur fréquence, et on leur donne un numéro de rang (1, 2, 3 …) alors la fréquence d'apparition Nσ(i) du n-uplet de rang i dans la suite est donnée par :

$$N_\sigma(i) = k \times i^{-\alpha}$$

Où k et α sont des constantes positives.

Cette distribution se représente graphiquement en échelle bi-logarithmique, avec en abscisse le rang des motifs et en ordonnée leur fréquence d'apparition. Cette représentation sera appelée courbe de Zipf, le motif de distribution suivant la loi de Zipf est une ligne droite.

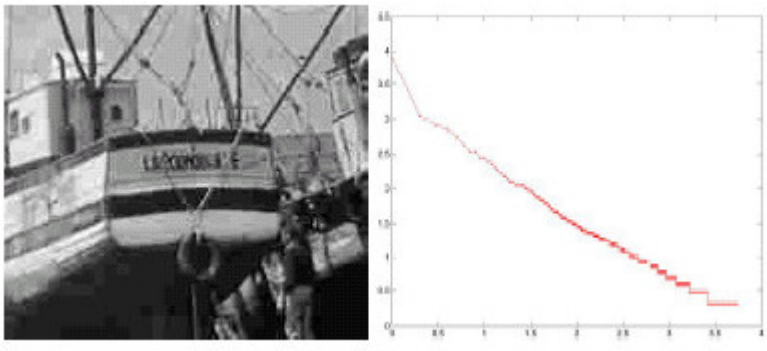


Figure 53 Loi de Zipf sur une image

Dans le cas des images, un motif peut être défini comme une matrice carrée de taille m x m de pixels adjacents de l'image. Comme on peut le voir dans l'image ci-dessus, l'organisation de ces motifs suit bien une loi de Zipf.

Dans le cas des textes en langage naturel, la valeur de l'exposant α est proche de 1.On a donc :

$$N_\sigma(i) \times i = cste$$

Ainsi pour le vocabulaire anglais

Rang	Mot	Fréquence	Rang*Fréquence
1	the	69 971	69 971
2	of	36 411	72 822
3	and	28 852	86 556
4	to	26 149	104 596
5	a	23 237	116 185
6	in	21 341	128 046
7	that	10 595	74 165

Tableau 59 Répartition du vocabulaire anglais

En recherche d'information, le vocabulaire utilisé dans les documents suit la loi de Zipf. Le produit de la fréquence d'un descripteur par son rang est approximativement constant, quel que soit le descripteur considéré dans un ensemble de documents donnés. Un autre type d'analyse peut être effectué sur le texte ; cette analyse est parfois appelée Zipf inverse. Elle consiste en l'analyse de la distribution des fréquences des mots, essentiellement ceux qui ont une fréquence faible. Zipf considère que ces fréquences suivent elles aussi une loi de puissance : le nombre de mots distincts $I(f)$ qui ont la fréquence *f,* est déterminé par la relation :

$I(f) = l \times f^{-\beta}$ Où l et β sont des constantes positives.

De cette manière, si on trace la courbe des $I(f)$ en échelle double logarithmique, comme pour la loi précédente, celle-ci doit se rapprocher d'une droite. Zipf estime le coefficient aux alentours de 2 pour le texte. Ces lois et plus précisément la première sont utilisées en recherche d'information pour déterminer les mots qui représentent le mieux le contenu d'un document. Pour cela, un concept supplémentaire est introduit : la conjecture de Luhn.

2 Conjecture de Luhn

La conjecture de Luhn émet une hypothèse sur l'information contenue dans les termes d'un document. Elle considère que les descripteurs non pertinents sont les descripteurs de rangs faibles (très fréquents) car ce sont des mots qui reviennent souvent, ils n'ont pas de pouvoir discriminant. Par exemple, dans un corpus de documents informatiques, le mot ordinateur est très fréquent et ne permet pas de différencier les documents entre eux. La conjecture de Luhn considère aussi ceux de rangs élevés (très rares) comme peu pertinents, en effet ces mots sont rares et donc peu utilisés. Les descripteurs qui sont pertinents sont les descripteurs de rangs intermédiaires.

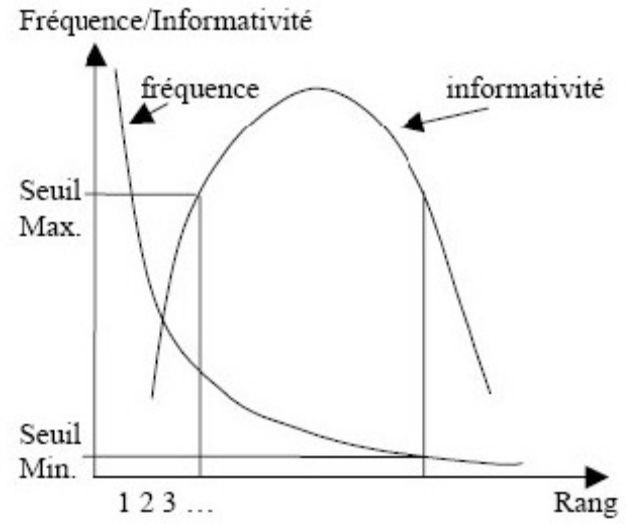


Figure 54 Conjecture de Luhn

Les systèmes de recherche d'information utilisent cette conjecture pour diminuer la taille des index des documents. Le système fixe deux seuils de fréquence pour éliminer les termes dont le contenu informatif est faible. La représentation des documents n'utilise que les termes entre ces deux seuils.

3 Utilisabilité des concepts

Nous testons, dans un premier temps, si les concepts suivent ces deux lois et s'il s'agit par conséquent de descripteurs utilisables en recherche d'information. Pour cela nous traçons d'une part la courbe du cumul des fréquences des termes avec les termes classés par ordre descendant de fréquence, nous comparons les courbes obtenues pour les concepts avec une courbe obtenue pour les lemmes. Les résultats de la figure 55 montrent des courbes très similaires et ces trois courbes montrent que, dans les quatre cas, environ 10 % des descripteurs représentent 80% du vocabulaire. Nous pouvons donc convenir que les concepts suivent bien la loi de Zipf. Cela est confirmé par la courbe qui décrit le pourcentage de vocabulaire couvert en fonction du pourcentage de descripteurs, où les quatre courbes ont la même forme. Cette forme est proche d'une droite mais un peu bombée, avec deux courbes similaires pour l'extraction MetaMap et les lemmes, et deux courbes similaires MapMiniPar et MapTreeTagger. Cela signifie d'une part que MapMiniPar et MapTreeTagger fournissent des extractions très poches, c'est que qui a déjà était vu sur le comptage des descripteurs. D'autre part, l'indexation MetaMap fournit un nombre de concepts plus élevé que les deux autres méthodes et un nombre de descripteurs équivalent au nombre de lemmes, elle fournit par conséquent une courbe proche de celle des lemmes.

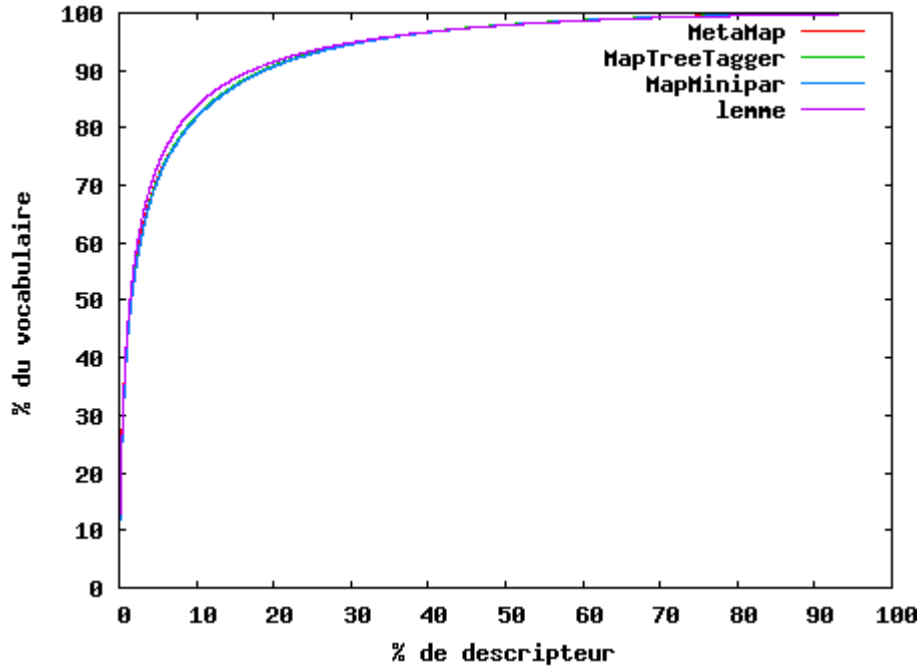


Figure 55 Courbe de cumul des concepts extraits par les trois méthodes de génération des concepts

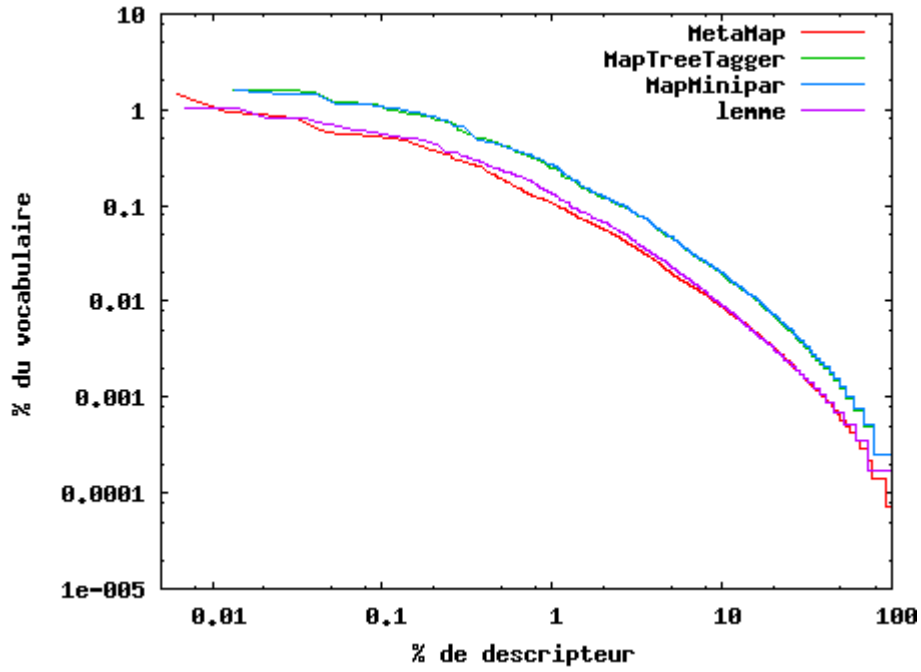


Figure 56 Pourcentage de vocabulaire ne fonction du pourcentage de descripteurs pour les concepts (échelle double logarithmique)

4 Utilisabilité des relations sémantiques

Dans un deuxième temps, nous testons si les concepts suivent les deux lois énoncées en recherche d'information et si par conséquent ce sont des descripteurs utilisables. Comme pour les concepts, nous

traçons la courbe des cumuls figure 57, ici les trois courbes sont assez similaires cependant contrairement aux concepts, les 10% des descripteurs les plus fréquents ne représentent que 60 %du vocabulaire. Cela vient du faut que les relations sont plus variées que les concepts. La figure décrivant le pourcentage de vocabulaires couverts en fonction du pourcentage de descripteurs montre deux courbes proches pour les méthodes MapMiniPar et MapTreeTagger et une légèrement différente pour MetaMap. Ces courbes ont bien des formes de droites mais on peut remarquer que l'inclinaison de ces courbes est plus faible que celle des courbes de concepts. Le descripteur le plus fréquent représente ici 0.1 % du vocabulaire alors qu'il en représente environ 1% sur les concepts. Les lois de Zipf permettant l'application de la conjecture de Luhn sont applicables, il est cependant nécessaire de prendre en compte la typicité des relations sémantiques lors de leur utilisation en recherche d'information.

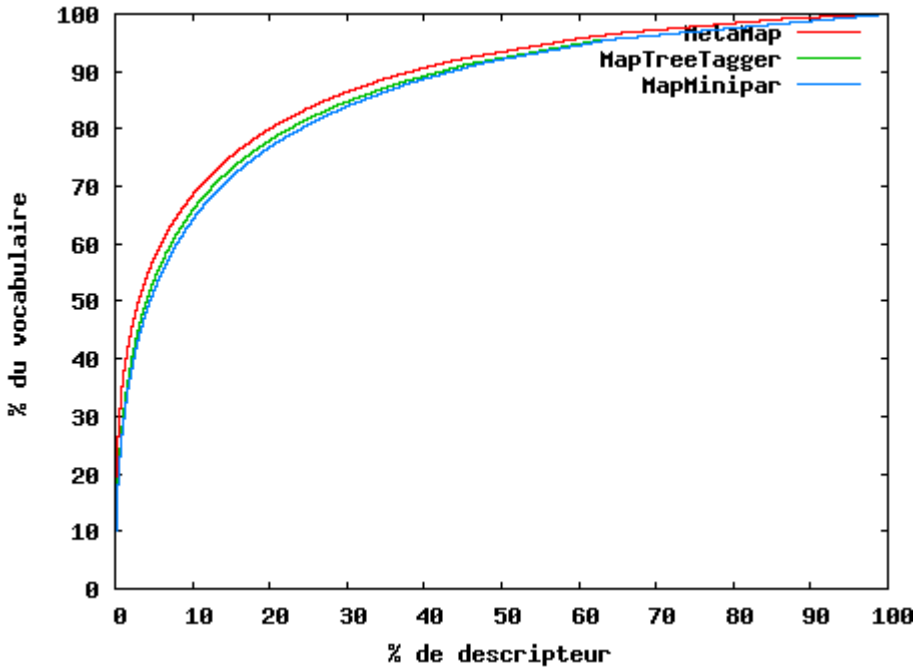


Figure 57 Courbe de cumul des relations extraites par les trois méthodes de génération des concepts

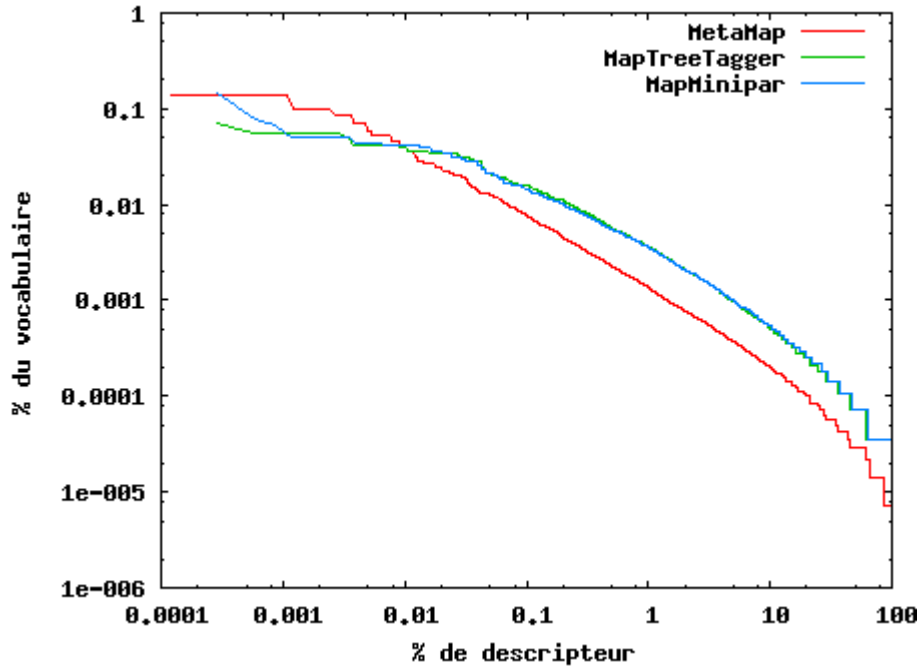


Figure 58 Pourcentage de vocabulaires en fonction du pourcentage de descripteurs pour les relations (échelle double logarithmique)

Annexe B. Applications des Modèles sur des Structures Syntaxiques

« En essayant continuellement on finit par réussir. Donc : plus ça rate, plus on a de chance que ça marche.» Jacques Rouxel (Extrait de la BD Les shadoks)

Nous proposons deux modèles de recherche d'information appliqués au texte. Ces modèles sont ensuite appliqués sur des textes médicaux. Nous proposons ici une version dégradée de ces modèles appliqués sur des analyses syntaxiques des textes. Dans ces modèles dégradés nous n'utilisons plus le type concepts mais le type lemmes, qui correspond à la version lemmatisée des mots extraits pas l'analyseur syntaxique et nous utilisons des relations syntaxiques entre ces lemmes à la place des relations sémantiques.

Nous voulons prouver que l'utilisation de nos modèles sur des analyses en dépendance permet d'augmenter la performance en recherche d'information, notamment dans des cas de recherches orientées vers la précision des réponses, donc le cas de requêtes précises. En effet les analyseurs syntaxiques sont plus facilement disponibles pour de nombreux domaine que des méthodes de création de graphes sémantiques.

5 Instanciation

5.1 Support de types

Le support de types est défini par l'analyseur syntaxique, il est constitué d'un ensemble de lemmes qui sont les lemmes reconnus par l'analyseur et par un ensemble de relations qui sont les relations syntaxiques détectées par l'analyseur en dépendance. Nous utilisons trois analyseurs syntaxiques pour trois langues différentes, l'analyse syntaxique ne permettant pas d'avoir une représentation multilingue des documents. Nous définissons donc les trois supports suivants :

- Pour le français, nous utilisons l'analyseur 'Xerox Incremental Parser' (XIP) (Aït-Mokhtar *et al.*, 2002)et nous définissons le support de types :

$$ST_{XIP} = \left(T_{lemmesXIP}, T_{depedencesXIP}, T_{reférent}\right)$$

avec $T_{lemmesXIP}$=*{lemme définis par XIP}*, $T_{dependencesXIP}$=*{dépendance définis par XIP }*, $T_{référents}$=*{numéro}*.

- Pour le finnois, nous utilisons l'analyseur Conexor (Tapanainen, 1999) et nous définissons le support de types :

$$ST_{CO} = (T_{lemmesCO}, T_{depedencesCO}, T_{référent})$$

avec $T_{lemmesCO}$={*lemme définis par conexor*}, $T_{dependencesCO}$={*dépendance définis par conexor*}, $T_{référents}$={*numéro*}.

- Pour le russe, nous utilisons l'analyseur ETAP (Apresian *et al.*, 2003) et nous définissons le support de types :

$$ST_{ET} = (T_{lemmesET}, T_{depedencesET}, T_{référent})$$

avec $T_{lemmesET}$={*lemme définis par ETAP*}, $T_{dependencesET}$={*dépendance définis par ETAP*}, $T_{référents}$={*numéro*}.

- Pour l'anglais, nous utilisons l'analyseur MiniPar et nous définissons le support de types :

$$ST_{MP} = (T_{lemmesMP}, T_{depedencesMP}, T_{référent})$$

avec $T_{lemmesMP}$={*lemme définis par MiniPar*}, $T_{dependencesMP}$={*dépendance définis par MiniPar*}, $T_{référents}$={*numéro*}.

5.2 Représentation intermédiaire

Dans ces trois langues nous obtenons la représentation intermédiaire du document en utilisant l'analyseur syntaxique adapté à la langue du texte à traiter. Le résultat de l'analyseur fournit un arbre de dépendance, dans lequel les nœuds sont des lemmes et les relations des relations syntaxiques. Cette représentation est donc un graphe, ce qui nous permet de générer les deux modèles proposés à partir de cette représentation.

6 Contexte et évaluation de la génération des arbres

Ne connaissant pas le comportement des éléments qui forment les arbres syntaxiques lorsqu'ils sont utilisés en recherche d'information, nous testons si ces descripteurs suivent les mêmes lois que les descripteurs habituellement employés. Pour cela nous étudions la répartition des lemmes et des dépendances extraites par l'analyseur syntaxique. Nous étudions ensuite les apports et les résultats d'un modèle basé sur l'arbre de dépendance par rapport à un modèle à base de lemmes et son efficacité en fonction de la langue des documents.

6.1 Description du corpus

Pour évaluer les dépendances, nous utilisons les corpus des campagnes CLEF 2002 et 2003 (Peters *et al.*, 2003). Ces corpus sont constitués de collections de journaux de différentes langues. Nous utilisons seulement trois de ces langues : le français, le finnois et le russe. Le détail de ces collections est décrit dans le tableau 60. La campagne CLEF 2002 contient un ensemble de 50 requêtes formées d'une phrase courte. La campagne 2003, quant à elle, contient 60 requêtes, chacune de ces requêtes est constituée de trois parties : un titre, une phrase courte et une description succincte du besoin d'informations, un exemple de requête est donné dans le Tableau 60 Description des collections

Dans cet article nous utilisons seulement le titre et la phrase de chaque requête. Pour une utilisation multilingue, l'ensemble des ces requêtes est traduit manuellement dans les différentes langues.

Langue	Journal	Année	Nb. de document
français	Le monde	1994	44 013
		1995	NA
	ATS	1994	43 178
		1995	42 615
finnois	Aamulehti	1994-1995	55 344
russe	Izvestia	1995	16 761

Tableau 60 Description des collections

Numéro de la requête	C146
Titre	Les Fast-foods au Japon
Description	Quelles chaînes de fast-food nord-américaines ont un grand nombre de restaurants au Japon ?
Narration	Les documents pertinents doivent mentionner le nom des chaînes de fast-food américaines qui rencontrent le plus de succès au Japon, et peuvent contenir des informations supplémentaires concernant l'introduction de ce type d'alimentation dans la société japonaise.

Tableau 61 Exemple de requête de CLEF 2003

6.2 Utilisabilité des dépendances

Nous testons par la suite si l'ensemble des dépendances extraites à partir de documents français suit la loi de Zipf. Pour cela nous utilisons une partie du corpus constituée de la collection française 'Le monde' de 1994, sur laquelle nous extrayons un certain nombre de descripteurs :

- les mots qui constituent les documents sans appliquer d'autres traitements;
- les lemmes extraits par l'analyseur XIP;
- les dépendances extraites par l'analyseur XIP.

	lemme	mot	dépendance
descripteur différent	185 848	215 751	5 839 732
Nombre d'occurrences des descripteurs	15 299 359	21 859 214	26 264 464

Tableau 62 Données sur les descripteurs utilisés

Comme on peut le voir sur le tableau 62, si le nombre de dépendances extraites de la collection est proche du double du nombre de lemmes, le nombre de dépendances différentes est environ 30 fois supérieur au nombre de lemmes différents. Même si l'importance de ce nombre n'est pas négligeable, notamment pour la taille des index, il reste acceptable par rapport au nombre de dépendances théoriquement possibles.

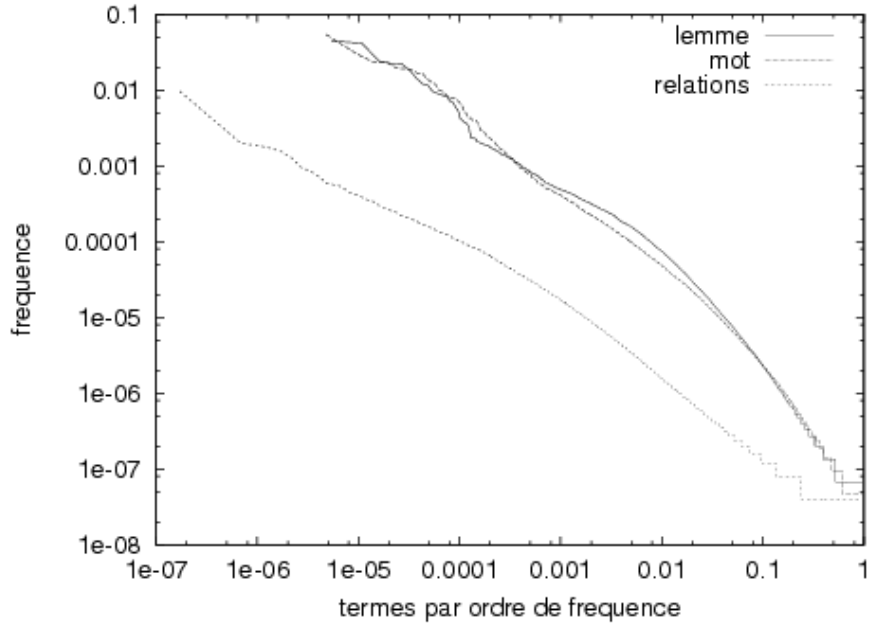


Figure 59 Répartitions de la fréquence des termes par leur rang

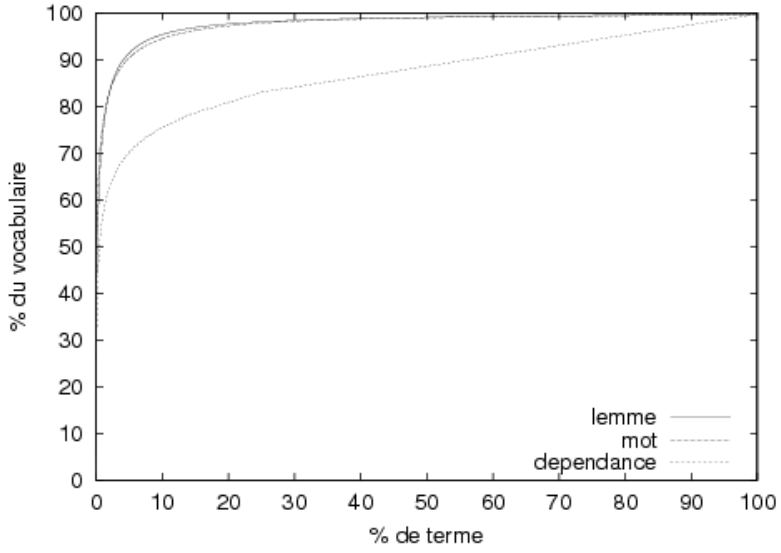


Figure 60 Répartition du vocabulaire sur la collection

Nous étudions ensuite la répartition des termes sur la collection. La figure 59 montre que la répartition des dépendances suit bien une loi de Zipf mais que le coefficient directeur de cette répartition est plus faible que pour celui des lemmes ; en effet, le calcul du coefficient directeur de la droite de régression pour les dépendances donne un résultat proche de 0.7 alors que pour les lemmes le coefficient est proche de 1.5. On en conclut que la répartition des dépendances est plus uniforme que celle des lemmes. Ce phénomène est confirmé sur la figure 60 où 10 % des lemmes représentent plus de 90% du vocabulaire de la collection alors que 10% des dépendances n'en représentent que 75%.

Il est aussi important de souligner, sur la figure 60, l'importance du nombre de dépendances n'apparaissant qu'une seule fois dans le corpus. Un processus de recherche d'information utilisant les dépendances s'avère ainsi fortement sélectif.

A condition de prendre en compte certaines des spécificités des dépendances lors de l'indexation (tel que le nombre de relations n'apparaissant qu'une fois), Nous pouvons par la suite utiliser ces dépendances pour une tâche de recherche d'information.

7 Modèle Local

7.1 Vocabulaire de lemmes et de dépendances

Dans cette partie nous utilisons les différents corpus de CLEF 2003 pour évaluer une modélisation à base de dépendances. Nous testons dans un premier temps cette indexation sur le corpus français puis nous effectuons des tests similaires avec les deux autres langages.

Une expérience ayant une pondération, ne prenant en compte que le nombre d'occurrences d'un descripteur dans un document *(tf)*, produit la courbe de rappel précision de la figure 61. Une deuxième expérimentation, en utilisant une pondération à base de *tf-idf*, produit la courbe de la figure 62.

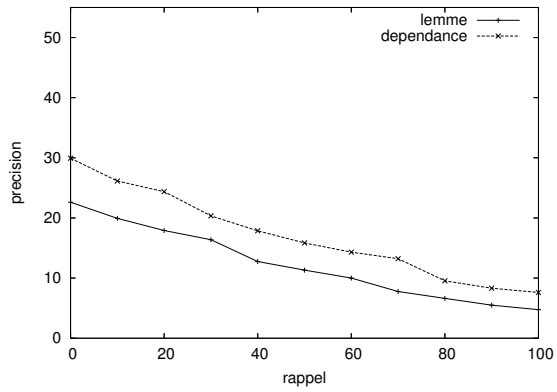


Figure 61 Courbe de rappel précision sur le corpus français de CLEF 03 avec tf

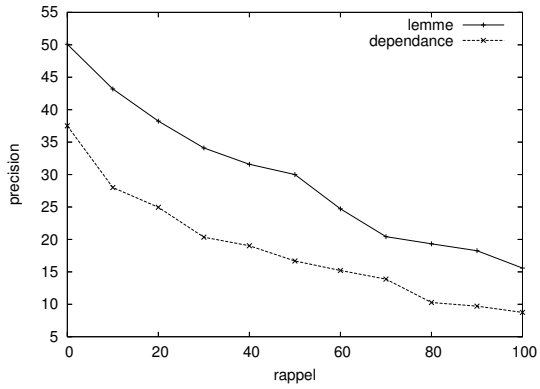


Figure 62 Courbe de rappel précision sur le corpus français de CLEF 03 avec tf.idf

À l'aide du tf, les résultats obtenus avec l'index basé sur les dépendances sont meilleurs que les résultats avec l'index basé sur les lemmes. L'utilisation d'une pondération, tel que *tf.idf* sur ces descripteurs, inverse les résultats. En effet, avec l'utilisation de cette pondération, le résultat des lemmes est fortement amélioré alors que celui des dépendances n'est pas amélioré dans la même proportion. Cela signifie qu'une pondération à base de *tf.idf* n'est pas suffisamment adaptée aux dépendances et c'est une pondération pour les mots. Cette observation est renforcée par les résultats sur les deux autres langues (cf. tableau 63) où l'utilisation du *tf.idf* sur les dépendances ne modifie pas les résultats et qui, de plus, dans le cas du finnois les fait baisser. Le tableau montre aussi que l'utilisation d'autres pondérations telles que la 'divergence from randomness' *DFR* ont le même effet que le *tf.idf*, l'amélioration apportée aux lemmes est largement supérieure que celle apportée aux dépendances.

Langue	Type de pondération	Lemme	Dépendance
Français	Tf	0.1155	0.1614
	Tf.idf	0.2841	0.1757
	DFR	0.4130	0.2114
Finnois	Tf	0.0911	0.0906
	Tf.idf	0.2357	0.0841
	DFR	0.3846	0.1003
Russe	Tf	0.0601	0.0729
	Tf.idf	0.1433	0.1035
	DFR	0.2740	0.1701

Tableau 63 Résultat en précision moyenne pour les trois langues

Un certain nombre des dépendances extraites des requêtes ne sont pas présentes dans l'index des documents. Cela vient en partie du grand nombre de dépendances différentes et de la quantité de celles n'apparaissant qu'une fois. Ce phénomène est cependant accentué par la structure des requêtes utilisées dans la campagne de CLEF qui sont sous la forme de questions ou de phrases telles que : 'trouver des documents qui…'. Elles décrivent donc un besoin d'information et leur structure est différente de celle des documents à rechercher. Par conséquent, cela diminue les performances du système. Un traitement particulier des requêtes brutes, telles qu'elles apparaissent dans la collection de test, est donc nécessaire pour éviter cet artefact.

Au final, un certain nombre de requêtes sont améliorées et donnent de meilleurs résultats par l'utilisation des dépendances plutôt que des lemmes. Sur cette collection, 14 requêtes donnent de meilleurs résultats avec les dépendances, et pour la majorité de celles-ci, la précision à 5 documents est améliorée. Cela signifie donc que dans *certains cas*, l'utilisation de dépendances peut permettre d'améliorer la précision d'un système de recherche d'information.

7.2 Modèle complet

Puisque dans certains cas les dépendances améliorent les résultats de recherche d'information, nous combinons ces deux descripteurs au sein de notre modèle local dans le but de tirer profit des spécificités des deux descripteurs. Nous pondérons les vocabulaires à l'aide d'un *tf.idf*. Les résultats du modèle local sont présentés sur le tableau 64, ils montrent que l'utilisation du modèle local permet d'améliorer les résultats. Cependant l'amélioration varie beaucoup en fonction des langues. En effet, les résultats en russe sont fortement améliorés (28%) alors que ceux en français ne le sont que beaucoup plus faiblement (3%).

	français	Russe	finnois
Résultat lemme	0.2841	0.1433	0.2357
Résultat dépendance	0.1757	0.1035	0.1003
Résultat regroupement	0.2939	0.1916	0.264

Tableau 64 Résultat en précision moyenne du regroupement des résultats en tf.idf sur les trois langues

L'utilisation des lemmes et des relations syntaxiques permet donc d'améliorer les résultats de recherche d'information. Cependant ces résultats restent inférieurs aux résultats obtenus par des modèles à base de lemmes utilisant des modèles évolués tels que *DFR*.

8 Modèle Global

Pour tester le modèle global sur les arbres syntaxiques, nous utilisons la collection CLEF médicale. Au niveau syntaxique nous ne testons que le modèle global sans étiquettes. Nous divisons les requêtes de CLEF médicale 2005 et 2006 comme présenté dans l'évaluation de modèle global au niveau sémantique. Le modèle proposé est de plus comparé avec le modèle proposé par GAO (Gao *et al.*, 2004) détaillé dans l'état de l'art. Nous obtenons les résultats présentés sur le tableau 65pour la précision moyenne et les résultats présentés sur le tableau 66pour la précision à 5 documents[28].

	$\lambda_{concept}$	$\lambda_{relation}$	entraînement	évaluation
GAO	0.8	0.9	0.194	**0.210**
MG	0.4	0.9	0.218	0.209
ML	0.4		**0.220**	0.209

Tableau 65 Résultats en précision moyenne pour le modèle global sans étiquettes sur MapMiniPar

	$\lambda_{concept}$	$\lambda_{relation}$	entraînement	évaluation
GAO	0.2	0.9	0.288	0.287
MG	0.2	0.9	**0.344**	**0.300**
ML	0.2		0.336	0.287

Tableau 66 Résultats en précision à 5 documents pour le modèle global sans étiquettes sur MapMiniPar

Sur cette collection, comme pour les concepts, le modèle de GAO et le modèle utilisé ici sont très proches, quelque soit la méthode de calcul des estimations. Il n'y a pas de différence significative entre les deux, de même qu'il n'y a pas de différence significative entre ces deux modèles et celui basé uniquement sur les unigrammes. Pour la précision moyenne, le modèle global donne des résultats très légèrement supérieurs. S'il n'y a pas de différence significative entre le modèle à base de concepts et le modèle local, l'utilisation de relations améliore la précision à 5 documents. Les relations sont assez rares dans les documents ce qui permet d'améliorer la précision quand celles-ci sont détectées, cependant la correspondance des relations sans étiquette est proche d'un modèle de cooccurrence des termes au sein des phrases.

[28] Pour la sélection des paramètres notamment pour la P5D en cas d'égalité, le paramètre le plus faible est sélectionné

9 Conclusion

Nous montrons d'une part, que la loi de Zipf est toujours respectée pour les dépendances, et d'autre part, que l'utilisation des dépendances syntaxiques en tant que descripteurs dans une tâche de recherche d'information, donne des résultats encourageants. En effet, dans certains cas (qui restent à analyser), on a pu observer que les dépendances permettent d'améliorer la recherche d'information et d'atteindre une meilleure précision.

Du fait de leur répartition particulière face aux termes, et du fait qu'elles sont constituées de lemmes, les dépendances n'ont pas le même comportement que les autres descripteurs face aux modèles de pondération standard. Elles sont donc des descripteurs particuliers nécessitant une pondération adaptée, notamment par rapport aux spécificités de leur répartition sur le corpus. Cependant même la prise compte ces particularités comme le fait le modèle global, ne permet pas d'améliorer les performances du système.

En parallèle, un facteur essentiel à prendre en compte est la performance de l'analyseur syntaxique et la qualité des dépendances extraites. Les variations des performances sur les différentes langues peuvent provenir de la structure même des langues mais aussi de la qualité des informations extraites par les analyseurs syntaxiques. Étudier les variations de performance entre différents analyseurs sur une même langue pourrait être profitable, de même qu'intégrer dans une pondération finale une mesure de la certitude de l'existence d'une dépendance(Brunet-Manquat, 2004).

Les dépendances montrent des lacunes, en effet, elles montrent seulement de faibles améliorations. Elles portent surtout une information sur la structure de la phrase. Si dans certains cas cette structure apporte de l'information, en recherche d'information cette information est souvent déjà partiellement prise en compte par la cooccurrence des lemmes. La vision des documents à travers des lemmes et des relations syntaxiques n'est pas suffisamment complémentaire en recherche d'information, cela peut être différent pour d'autres types de tâches (Maisonnasse et Tambellini, 2007). Ces expérimentations montrent la nécessité d'utiliser un niveau de représentation plus sémantique.

Annexe C. Représentations d'un document

Nous présentons ici la représentation d'un document à l'aide des trois types de représentations présentées dans cette thèse. Pour cela nous utilisons un formalisme XML qui contient les informations sur le document suivant :

```
<Title>LIVER-BILIARY</Title>
<Description>
LIVER-BILIARY: Case# 127: LIVER LACERATION, DELAYED HEMORRHAGE.
This is a follow up examination from a MVA the previous day.
1) Active extravasation into the peritoneal cavity secondary to liver laceration.
A marked amount of fluid/blood is seen throughout the peritoneal cavity which has increased greatly seen the previous examination.
</Description>
```

1 Représentation intermédiaire

Dans cette représentation chaque phrase est représentée. Nous présentons dans la suite la représentation XML de la représentation intermédiaire du document. Les balises *LUNIT* séparent les phrases, les balises *CON* listent les concepts, et les balises *REL* les relations. Dans les balises *CON* l'attribut *name* donne le nom du concept identifié par *id*. Dans les relations *REL* l'attribut *name* donne le nom de l'étiquette qui relie les concepts identifiés par *C0* et *C1*. Que ce soit pour les concepts ou pour les relations, la représentation intermédiaire fournit deux scores : la fréquence et la confiance qui correspondent respectivement aux balises *freq* et *conf*.

```
<Description>
<LUNIT>
   <CON id="C0023884" name="Gastrointestinal Tract, Liver" TS0="T023" freq="1" conf="1" />
   <CON id="C0043246" name="Tear" TS0="T037" freq="1" conf="1" />
   <CON id="C0019080" name="Hemorrhage, unspecified" TS0="T033" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0043246" name="T135" val="location_of" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0043246" name="T146" val="disrupts" freq="1" conf="1" />
   <REL C0="C0019080" C1="C0043246" name="T150" val="manifestation_of" freq="1" conf="1" />
   <REL C0="C0019080" C1="C0043246" name="T166" val="associated_with" freq="1" conf="1" />
</LUNIT>
<LUNIT>
     <CON id="C0031809" name="Medical Inspection" TS0="T058" freq="1" conf="1" />
</LUNIT>
<LUNIT>
   <CON id="C0015376" name="Spillage"  TS0="T046" freq="1" conf="1" />
```

```
    <CON id="C0031138" name="Cavity of greater sac"  TS0="T030" freq="1" conf="1" />
    <CON id="C1522484" name="metastatic"  TS0="T081" freq="1" conf="1" />
    <CON id="C0023884" name="Gastrointestinal Tract, Liver"  TS0="T023" freq="1" conf="1" />
    <CON id="C0043246" name="Tear" TS0="T037" freq="1" conf="1" />
    <REL C0="C0015376" C1="C0031138" name="T135" val="location_of" freq="1" conf="1" />
    <REL C0="C0031138" C1="C1522484" name="T182" val="measurement_of" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0031138" name="T135" val="location_of" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0031138" name="T173" val="adjacent_to" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0031138" name="T174" val="connected_to" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0031138" name="T175" val="interconnects" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0031138" name="T133" val="part_of" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0031138" name="T134" val="contains" freq="1" conf="1" />
    <REL C0="C0015376" C1="C0023884" name="T135" val="location_of" freq="1" conf="1" />
    <REL C0="C0031138" C1="C0043246" name="T135" val="location_of" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0043246 name="T135" val="location_of" freq="1" conf="1" />
    <REL C0="C0023884" C1="C0043246 name="T146" val="disrupts" freq="1" conf="1" />
    <REL C0="C0015376" C1="C0043246" name="T149" val="complicates" freq="1" conf="1" />
    <REL C0="C0015376" C1="C0043246"name="T157" val="result_of" freq="1" conf="1" />
    <REL C0="C0015376" C1="C0043246"name="T137" val="co-occurs_with" freq="1" conf="1" />
    <REL C0="C0015376" C1="C0043246"name="T150" val="manifestation_of" freq="1" conf="1" />
    <REL C0="C0015376" C1="C0043246"name="T152" val="occurs_in" freq="1" conf="1" />
</LUNIT>
<LUNIT>
    <CON id="C0031138" name="Cavity of greater sac"  TS0="T030" freq="1" conf="1" />
    <CON id="C0031809" name="Medical Inspection"  TS0="T058" freq="1" conf="1" />
</LUNIT>
</Description>
```

2 Représentation du document Modèle Local

Dans cette représentation le document est représenté par un ensemble de concepts et un ensemble de relations. Ces deux ensembles synthétisent les représentations de phrase. Par conséquent cette représentation utilise le même format XML pour les concepts et les relations.

```
<Concepts>
    <CON id="C0043246" name="Tear" TS0="T037" freq="2" conf="2" />
    <CON id="C0019080" name="Hemorrhage, unspecified" TS0="T033" freq="1" conf="1" />
    <CON id="C0031809" name="Medical Inspection" TS0="T058" freq="2" conf="2" />
    <CON id="C0015376" name="Spillage"  TS0="T046" freq="1" conf="1" />
    <CON id="C0031138" name="Cavity of greater sac"  TS0="T030" freq="2" conf="2" />
    <CON id="C1522484" name="metastatic" TS0="T081" freq="1" conf="1" />
    <CON id="C0023884" name="Gastrointestinal Tract, Liver" TS0="T023" freq="2" conf="2" />
</Concepts>
```

```
<relations>
     <REL C0="C0015376" C1="C0031138" name="T135" val="location_of" freq="1" conf="1" />
   <REL C0="C0031138" C1="C1522484" name="T182" val="measurement_of" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0031138" name="T135" val="location_of" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0031138" name="T173" val="adjacent_to" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0031138" name="T174" val="connected_to" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0031138" name="T175" val="interconnects" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0031138" name="T133" val="part_of" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0031138" name="T134" val="contains" freq="1" conf="1" />
   <REL C0="C0023884" C1="C0043246" name="T135" val="location_of" freq="2" conf="2" />
   <REL C0="C0023884" C1="C0043246" name="T146" val="disrupts" freq="2" conf="2" />
   <REL C0="C0019080" C1="C0043246" name="T150" val="manifestation_of" freq="1" conf="1" />
   <REL C0="C0019080" C1="C0043246" name="T166" val="associated_with" freq="1" conf="1" />
   <REL C0="C0015376" C1="C0023884" name="T135" val="location_of" freq="1" conf="1" />
   <REL C0="C0031138" C1="C0043246" name="T135" val="location_of" freq="1" conf="1" />
   <REL C0="C0015376" C1="C0043246" name="T149" val="complicates" freq="1" conf="1" />
   <REL C0="C0015376" C1="C0043246"name="T157" val="result_of" freq="1" conf="1" />
   <REL C0="C0015376" C1="C0043246"name="T137" val="co-occurs_with" freq="1" conf="1" />
   <REL C0="C0015376" C1="C0043246"name="T150" val="manifestation_of" freq="1" conf="1" />
   <REL C0="C0015376" C1="C0043246"name="T152" val="occurs_in" freq="1" conf="1" />
</relations>
```

3 Représentation du document Modèle Global

Dans cette représentation le document est représenté par un ensemble de concepts, un ensemble de couples et un ensemble de relations sur tout le vocabulaire. Ces deux ensembles synthétisent les représentations de phrases par un modèle de graphe. Ici le format XML utilise l'attribut *P* pour donner la probabilité des éléments dans le modèle de graphe. Nous rajoutons la balise COU qui désigne le couple formé des concepts désignés par C0 et C1.

```
<Concepts>
   <CON id="C0019664" name="Histories" TS0="T090" p="0.0035"/>
   <CON id="C0030193" name="sense of pain" TS0="T184" p="0.0531"/>
   <CON id="C0234254" name="Pain, Radiating" TS0="T184" p="0.0531"/>
   <CON id="C0004600" name="Dorsum" TS0="T029" p="0.0085"/>
   ...
   <CON id="C0043246" name="Tear" TS0="T037" P="0.1745" />
   <CON id="C0019080" name="Hemorrhage, unspecified" TS0="T033" P="0.0923" />
   <CON id="C0031809" name="Medical Inspection" TS0="T058" P="0.1542" />
   <CON id="C0031138" name="Cavity of greater sac" TS0="T030" P="0.1920" />
   <CON id="C1522484" name="metastatic" TS0="T081" P="0.1506" />
   <CON id="C0023884" name="Gastrointestinal Tract, Liver" TS0="T023" P="0.1542" />
   ...
</Concepts>
```

```
<couples>
   <COU C0="C0015780" C1="C0019664" P="0.0015" >
   <COU C0="C0019270" C1="C0019664" P="0.00824" >
   <COU C0="C0019664" C1="C0817096" P="0.00158" >
   <COU C0="C0015780" C1="C0024141" P="0.00984" >
   ...
   <COU C0="C0015376" C1="C0031138" P="0.0258" />
   <COU C0="C0031138" C1="C1522484" P="0.0082" />
   <COU C0="C0023884" C1="C0031138" P="0.00892" />
   <COU C0="C0019080" C1="C0043246" P="0.06895" />
   <COU C0="C0015376" C1="C0023884" P="0.00685" />
   <COU C0="C0031138" C1="C0043246" P="0.0256" />
   <COU C0="C0015376" C1="C0043246" P="0.0368" />
   ...
</couples>
<relations>
   <REL C0="C0019270" C1="C0234254" name="T150" val="manifestation_of" P="0.6" />
   <REL C0="C0019270" C1="C0234254" name="T161" val="evaluation_of" P="0.6" />
   <REL C0="C0019270" C1="C0234254" name="T166" val="associated_with" P="0.6" />
   <REL C0="C0019664" C1="C0234254" name="T165" val="issue_in" P="0.6" />
   <REL C0="C0019664" C1="C0234254" name="T165" val="issue_in" P="0.6" />
   <REL C0="C0024141" C1="C0234254" name="T150" val="manifestation_of" P="0.6" />
   ...
   <REL C0="C0031138" C1="C1522484" name="T182" val="measurement_of" P="1" />
   <REL C0="C0023884" C1="C0031138" name="T135" val="location_of" P="1" />
   <REL C0="C0023884" C1="C0031138" name="T173" val="adjacent_to" P="1" />
   <REL C0="C0023884" C1="C0031138" name="T174" val="connected_to" P="0.6"  />
   <REL C0="C0023884" C1="C0031138" name="T175" val="interconnects" P="1"  />
   <REL C0="C0023884" C1="C0031138" name="T133" val="part_of" P="1" />
   <REL C0="C0023884" C1="C0031138" name="T134" val="contains" P="1" />
   <REL C0="C0023884" C1="C0043246" name="T135" val="location_of" P="1" />
   <REL C0="C0023884" C1="C0043246" name="T146" val="disrupts" P="0.6" />
   <REL C0="C0019080" C1="C0043246" name="T150" val="manifestation_of" P="1" />
   <REL C0="C0019080" C1="C0043246" name="T166" val="associated_with" P="1" />
   <REL C0="C0015376" C1="C0023884" name="T135" val="location_of" P="1" />
   <REL C0="C0015376" C1="C0043246" name="T149" val="complicates" P="1" />
   <REL C0="C0015376" C1="C0043246"name="T157" val="result_of" P="1" />
   <REL C0="C0015376" C1="C0043246"name="T137" val="co-occurs_with" P="1" />
   <REL C0="C0015376" C1="C0043246"name="T150" val="manifestation_of" P="1" />
   <REL C0="C0015376" C1="C0043246"name="T152" val="occurs_in" P="1" />
   ...
</relations>
```

LEXIQUE DES NOTATIONS

Par ordre alphabétique

Notation	Nom	Description	Page
B	Besoin d'informations	Motivation de l'activité de recherche	Page 74
C	Corpus de documents	Ensemble de documents	Page 74
$Conf()$	Calcul de la confiance finale	Calcul permettant d'obtenir un score de confiance pour une relation sémantique et un chemin syntaxique	Page 134
C_Q	Ensemble de concepts	Ensemble de concepts constituant la requête du modèle global	Page 100
CI	Corpus indexé	Ensemble des documents indexés	Page 74
ch	Chemin syntaxique	Ensemble des lemmes et des relations syntaxiques constituant le chemin qui relie deux lemmes dans l'arbre de dépendance d'une phrase	Page 133
DV	Sous-ensemble de vocabulaires de document	Sous-ensemble de vocabulaires défini par la sélection dans le vocabulaire correspondant des éléments représentant le document	Page 76
$DV_{concepts}^{IC}$	Ensemble des concepts d'un document selon IC	Ensemble des concepts détectés dans les phrases du document	Page 148
$DV_{concepts}^{MG}$	Ensemble des concepts probables d'un document selon MG	Ensemble des concepts probables formant la représentation indexée d'un document utilisant le modèle global	Page 96
$DV_{concepts}^{ML}$	Ensemble des concepts pondérés d'un document selon ML	Ensemble des concepts pondérés formant la représentation indexée d'un document utilisant le modèle local	Page 87
$DV_{concepts}^{MLC}$	Ensemble des concepts d'une requête selon MLc	Ensemble des concepts détectés dans une requête	Page 166
$DV_{concepts}^{ph}$	Ensemble des concepts pondérés d'une représentation de phrase	Ensemble des concepts pondérés formant la représentation d'une phrase	Page 114
DV_{lemmes}^{IC}	Ensemble des lemmes d'un document selon IL	Ensemble des lemmes détectés dans les phrases du document	Page 149
$DV_{relations}^{MG}$	Ensemble des relations étiquetées probables d'un document selon MG	Ensemble des relations probables formant la représentation indexée d'un document utilisant le modèle global	Page 96
$DV_{relations}^{ML}$	Ensemble des relations pondérées d'un document selon ML	Ensemble des relations pondérées formant la représentation indexée d'un document utilisant le modèle local	Page 87
$DV_{relations}^{IR}$	Ensemble des relations d'un document selon IR	Ensemble des relations détectées dans les phrases du document	Page 154
$DV_{relations}^{ph}$	Ensemble des relations pondérées d'une représentation de phrase	Ensemble des relations pondérées formant la représentation d'une phrase	Page 114
$DV_{couples}^{MG}$	Ensemble des couples probables d'un document MG	Ensemble des couples probables formant la représentation indexée d'un document utilisant le modèle global	Page 96
$\delta()$	Degré de correspondance	Degré de correspondance entre deux unités de vocabulaire	Page 92

Notation	Nom	Description	Page
d	Document		Page 74
di	Document indexé	Représentation du document par le système de recherche d'information	Page 74
di_{int}	Représentation intermédiaire	Représentation intermédiaire d'un document composé de représentations de phrases	Page 112
$elem$	Lemme ou relation d'un chemin syntaxique		Page 134
$f_{concepts}()$	Fonction des poids de deux concepts	Fonction permettant d'établir le degré de correspondance entre deux concepts	Page 92
f_{np}	Vocabulaire non pondéré correspondant	Pour un vocabulaire pondéré ou pour un sous-ensemble de ce vocabulaire, retourne l'ensemble des vocabulaires simples ou complexes correspondant à ce vocabulaire	Page 70
$f_{relations}()$	Fonction des poids de deux relations	Fonction permettant d'établir le degré de correspondance entre deux relations	Page 92
fv	Fonction qui détermine le type	Détermine le type fv(i) du support ST utilisé par le $i^{éme}$ type du vocabulaire	Page 69
G_Q	Graphe étiqueté	Graphe étiqueté représentant la requête du modèle global	Page 100
IC	Modèle d'indexation conceptuelle	Modèle vectoriel permettant une indexation conceptuelle	Page 148
IL	Modèle d'indexation à base de lemmes	Modèle vectoriel permettant une indexation sur des lemmes	Page 149
$ind()$	Fonction d'indexation	Transpose un document en son indexation	Page 74
$Ind^{ph}_{concept}$	Fonction d'indexation des concepts d'une phrase	Fonction qui, à partir d'une phrase, fournit l'ensemble des nœuds concepts qui apparaissent dans cette phrase	Page 116
Ind^{ph}	Fonction d'indexation des phrases	Fonction qui lie une phrase avec sa représentation sous forme de graphe	Page 113
Ind^{mod}	Fonction d'indexation des modèles	Fonction qui, pour l'ensemble des représentations de phrases du document, établit le lien avec la représentation définie par le modèle de document	Page 113
$ind^{mod\ MG}$	Fonction d'indexation du modèle MG	Fonction qui traduit une représentation intermédiaire du document en représentation selon le modèle global	Page 120
$ind^{mod\ ML}$	Fonction d'indexation du modèle ML	Fonction qui traduit une représentation intermédiaire du document en représentation selon le modèle local	Page 118
$Ind^{ph}_{relation}$	Fonction d'indexation des relations d'une phrase	Fonction qui, à partir d'une phrase et d'un ensemble de concepts, fournit la représentation de la phrase contenant les concepts et les relations	Page 116
$inter()$	Fonction d'interprétation	Traduit un besoin d'information en une requête	Page 74
IR	Modèle d'indexation à base de relations	Modèle vectoriel permettant une indexation sur des relations	Page 154
$\lambda_{concept}$	Paramètre de lissage des concepts	Paramètre qui permet d'équilibrer la contribution de la collection et la contribution du document pour les concepts lors du calcul du modèle de document	Page 120
λ_{couple}	Paramètre de lissage des couples	Paramètre qui permet d'équilibrer la contribution de la collection et la contribution du document pour les couples lors du calcul du modèle de document	Page 121
$\lambda_{relation}$	Paramètre de lissage des relations	Paramètre qui permet d'équilibrer la contribution de la collection et la contribution du document pour les relations lors du calcul du modèle de document	Page 121
lem	Lemme d'un chemin syntaxique		Page 134
M	Modèle de recherche d'information		Page 73

Notation	Nom	Description	Page
M_D^g	Modèle de graphe du document	Modélisation statistique sous forme de graphe d'un document.	Page 94
MG	Modèle global	Modèle de recherche d'information correspondant au modèle global	Page 94
ML	Modèle local	Modèle de recherche d'information correspondant au modèle local	Page 86
MLC	Modèle d'indexation basé sur des concepts	Le modèle d'indexation unigramme sur les concepts	Page 166
M_{Rs}	Modèle de relation sémantique	Modélisation statistique des chemins syntaxiques correspondant à une relation sémantique	Page 133
nc	Taille du corpus	Nombre de documents formant le corpus	Page 74
np	Nombre de pondérations	Nombre de pondérations associées à un vocabulaire	Page 70
$np_{conceptDoc}$	Nombre de pondérations des concepts dans le modèle ML		Page 87
$np_{conceptPh}$	Nombre de pondérations des concepts dans les représentations de phrase		Page 114
$np_{conceptReq}$	Nombre de pondération des concepts dans les requêtes		Page 91
$np_{relationDoc}$	Nombre de pondération des relations		Page 87
$np_{relationPh}$	Nombre de pondérations des relations dans les représentations de phrase		Page 114
$np_{relationReq}$	Nombre de pondérations des relations dans les requêtes		Page 91
$np_{coupleDoc}$	Nombre de pondérations des couples dans les documents		Page 95
nph	Nombre de phrases d'un document		Page 112
nst	Nombre de types du support	Nombre de types utilisés par le support de types	Page 67
nsv	Nombre de vocabulaires	Nombre de vocabulaires utilisés par un support *SV*	Page 72
nt	Nombre de types d'un vocabulaire		Page 69
nu	Nombre d'unités de vocabulaire d'un DV ou QV	Cardinalité d'un sous-ensemble de vocabulaires	Page 76
nvd	Nombre de vocabulaires de SVD	Nombre de vocabulaires utilisés par le support de vocabulaires de document	Page 74
nvq	Nombre de vocabulaires de SVQ	Nombre de vocabulaires utilisés par le support de vocabulaires de requête	Page 74
P	Pondération	Réel	Page 70
Pert()	Fonction de pertinence	Fonction qui calcule la pertinence du document vis-à-vis de la requête	Page 78
$p_{confianvce}$	Pondération de la confiance	Représente la confiance dans le processus d'extraction d'un élément au sein de la représentation finale du document	Page 117
ph	Phrase d'un document		Page 112
phi	Représentation de phrase	Représentation d'une phrase dans la représentation intermédiaire d'un document	Page 112
p_{ri}	Pondération de recherche d'information	Représente l'importance de l'élément pour la recherche d'information	Page 117
QV	Sous-ensemble de vocabulaires de requête	Sous-ensemble de vocabulaires défini par la sélection dans le vocabulaire correspondant des éléments représentant la requête	Page 78
$QV_{concepts}^{IC}$	Ensemble des concepts d'une requête selon IC	Ensemble des concepts détectés dans une requête	Page 148

Notation	Nom	Description	Page
$QV_{concepts}^{MG}$	Ensemble des concepts d'une requête selon MG	Ensemble des concepts probables formant la représentation d'une requête utilisant le modèle global	Page 100
$QV_{concepts}^{ML}$	Ensemble des concepts pondérés d'une requête selon ML	Ensemble des concepts pondérés formant la représentation d'une requête utilisant le modèle local	Page 91
$QV_{concepts}^{MLC}$	Relation de correspondance du modèle MLC	Relation de correspondance basée sur la vraisemblance de la requête.	Page 166
QV_{lemmes}^{IC}	Ensemble des lemmes d'une requête selon IL	Ensemble des lemmes détectés dans une requête	Page 149
$QV_{relations}^{IR}$	Ensemble des relations d'une requête selon IR	Ensemble des relations détectées dans une requête	Page 154
$QV_{relations}^{MG}$	Ensemble des relations d'une requête selon MG	Ensemble des relations probables formant la représentation d'une requête utilisant le modèle global	Page 100
$QV_{relations}^{ML}$	Ensemble des relations pondérées d'une requête selon ML	Ensemble des relations pondérées formant la représentation d'une requête utilisant le modèle local	Page 91
q	Requête	Représentation du besoin d'information par le système de recherche d'information	Page 74
RC	Relation de correspondance	Relation qui relie des documents indexés à des requêtes	Page 74
RC_{IC}	Relation de correspondance du modèle IC	Relation de correspondance basée sur le produit scalaire	Page 148
RC_{IL}	Relation de correspondance du modèle IL	Relation de correspondance basée sur le produit scalaire	Page 149
RC_{IR}	Relation de correspondance du modèle IR	Relation de correspondance basée sur le produit scalaire	Page 154
RC_{MG}	Relation de correspondance du modèle MG	Relation de correspondance pour le modèle global	Page 94
RC_{ML}	Relation de correspondance du modèle ML	Relation de correspondance pour le modèle local	Page 86
RC_{MLC}	Vocabulaire des concepts du modèle MLC		Page 166
R_Q	Ensemble de relations	Ensemble de relations constituant la requête du modèle global	Page 100
Rs	Relation sémantique	Type de relation sémantique	Page 133
$Rsyn$	Relation syntaxique d'un chemin syntaxique		Page 134
ST	Support de types	Liste des types utilisés par un système de recherche d'information	Page 67
ST_{IL}	Support de types du modèle IL	Support de types qui définit les lemmes utilisés par le modèle IL	Page 149
ST_{UMLS}	Support de types basé sur UMLS	Support qui définit les concepts et les relations sémantiques en se basant sur UMLS	Page 128
SV	Support de vocabulaires	Liste de vocabulaires qu'utilise un système de recherche d'information	Page 72
SVD	Support de vocabulaires de document	Support de vocabulaires qui représente le modèle de document	Page 74
SVD_{IC}	Support de vocabulaires de document du modèle IC		Page 148
SVD_{IL}	Support de vocabulaires de document du modèle IL		Page 149
SVD_{IR}	Support de vocabulaires de document du modèle IR		Page 154

Notation	Nom	Description	Page
SVD_{MG}	Support de vocabulaires de document du modèle MG	Support de vocabulaires définissant le modèle de document pour le modèle global	Page 94
SVD_{ML}	Support de vocabulaires de document du modèle ML	Support de vocabulaires définissant le modèle de document pour le modèle local	Page 86
SVD_{MLC}	Support de vocabulaires de document du modèle MLC		Page 166
SVP	Support de vocabulaires des représentations de phrase	Support de vocabulaires définissant le modèle de représentation des phrases	Page 114
SVQ	Support de vocabulaires de requête	Support de vocabulaires qui représente le modèle de requête	Page 74
SVQ_{IC}	Support de vocabulaires de requête du modèle IC		Page 148
SVQ_{IL}	Support de vocabulaires de requête du modèle IL		Page 149
SVQ_{IR}	Support de vocabulaires de requête du modèle IR		Page 154
SVQ_{MG}	Support de vocabulaires de requête du modèle MG	Support de vocabulaires définissant le modèle de requête pour le modèle global	Page 94
SVQ_{ML}	Support de vocabulaires de requête du modèle ML	Support de vocabulaires définissant le modèle de requête pour le modèle local	Page 86
SVQ_{MLC}	Support de vocabulaires de requête du modèle MLC		Page 166
T	Type		Page 66
$T_{concepts}$	Type des concepts	Type définissant les noms de concepts, par exemple à l'aide d'UMLS	Page 66
$T_{référents}$	Type des référents	Type définissant des référents	Page 84
$T_{relations}$	Type des relations	Type définissant les noms des relations entre concepts	Page 67
t	Élément de type		Page 66
uv	Unité de vocabulaire		Page 68
V	Vocabulaire		Page 68
$V_{concepts}$	Vocabulaire des concepts	Vocabulaire définissant tous les concepts possibles	Page 85
$V^{MG}_{conceptsDoc}$	Vocabulaire pondéré des concepts des documents du modèle MG	Vocabulaire pondéré des concepts utilisés par le support de vocabulaires de document du modèle global	Page 95
$V^{ML}_{conceptsDoc}$	Vocabulaire pondéré des concepts des documents du modèle ML	Vocabulaire pondéré des concepts utilisés par le support de vocabulaires de document du modèle local	Page 87
$V^{IC}_{concepts}$	Vocabulaire des concepts du modèle IC		Page 148
$V^{MLC}_{concepts}$	Ensemble des concepts d'un document selon MLC	Ensemble des concepts détectés dans les phrases du document	Page 166
$V_{conceptsPh}$	Vocabulaire pondéré des concepts des représentations de phrase	Vocabulaire pondéré des concepts utilisés par le support de vocabulaires des représentations de phrase	Page 114
$V^{ML}_{conceptsReq}$	Vocabulaire pondéré des concepts des requêtes du modèle ML	Vocabulaire pondéré des concepts utilisés par le support de vocabulaires de requête du modèle ML	Page 91
$V_{couples}$	Vocabulaire des couples	Vocabulaire définissant tous les couples possibles	Page 85
$V^{MG}_{couplesDoc}$	Vocabulaire pondéré des couples des documents du modèle MG	Vocabulaire pondéré des couples utilisés par le support de vocabulaires de document du modèle global	Page 95
V^{IL}_{lemmes}	Vocabulaire des lemmes du modèle IL		Page 149
$V_{relations}$	Vocabulaire des relations	Vocabulaire définissant toutes les relations possibles	Page 85

Notation	Nom	Description	Page
$V^{MG}_{relationsDoc}$	Vocabulaire pondéré des relations des documents du modèle MG	Vocabulaire pondéré des relations utilisées par le support de vocabulaires de document du modèle global	Page 95
$V^{ML}_{relationsDoc}$	Vocabulaire pondéré des relations des documents du modèle ML	Vocabulaire pondéré des relations utilisées par le support de vocabulaires de document du modèle ML	Page 87
$V^{IR}_{relations}$	Vocabulaire des relations du modèle IR		Page 154
$V_{relationPh}$	Vocabulaire pondéré des relations des représentations de phrase	Vocabulaire pondéré des relations utilisées par le support de vocabulaires des représentations de phrase	Page 114
$V^{ML}_{relationsReq}$	Vocabulaire pondéré des relations des requêtes du modèle ML	Vocabulaire pondéré des concepts utilisés par le support de vocabulaires de requête du modèle ML	Page 91

BIBLIOGRAPHIE

Alvarez C, Langlais P & Nie J. « Mots composés dans les modèles de langue pour la recherche d'information ». *11è édition de la conférence sur le Traitement Automatique des Langues Naturelles (TALN)* (2004) : pp. 11-16.

Amati G & Ounis I. « Conceptual Graphs and First Order Logic ». *The Computer Journal* (2000) 43: pp. 1-12.

Amati G & van Rijsbergen CJ. « Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness ». *ACM Transaction on Information Systems* (2002) 20: pp. 357-389.

Apresian J, Boguslavsky I, Iomdin L, Lazursky A, Sannikov V, Sizov V & Tsinman L. « ETAP-3 Linguistic Processor: a Full-Fledged NLP Implementation of the MTT ». *Proceeding of MTT* (2003) : pp. 279-288.

Aronson AR. « Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program ». *AMIA 2001 Annual Symposium on biomedical and health informatics* (2001) : pp. 17-27.

Aronson AR, Rindflesch TC, browne AC. « Exploiting a Large Thesaurus for Information Retrieval ». *RIAO 94 : recherche d'information assistée par ordinateur* (1994) : pp. 197-216.

Aït-Mokhtar S, Chanod J & Roux C. « Robustness beyond shallowness: incremental deep parsing ». *Natural Language Engineering* (2002) 8: pp. 121-144.

Baziz M. *Indexation Conceptuelle Guidée par Ontologie pour la Recherche d'Information*. Thèse de doctorat, université paul sabatier. 2005.

Baziz M, Boughanem M & Aussenac-Gilles N. « Conceptual Indexing Based on Document Content Representation ». *Lecture Notes in Computer Science* (2005) 3507: pp. 171-186.

Belkin NJ. « Anomalous states of knowledge as a basis for information retrieval ». *Canadian Journal of Information Science* (1980) 5: pp. 133-143.

Berger A & Lafferty JD. « Information Retrieval as Statistical Translation ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (1999) : pp. 222-229.

Berrut C. *Une méthode d'indexation fondée sur l'analyse sémantique de documents spécialisés. Le prototype RIME et son application à un corpus médical*. Thèse de doctorat, Université Joseph Fourier. 1988.

Boughanem M, Kraaij W & Nie J. « Modèles de langue pour la recherche d'information ». *Les systèmes de recherche d'informations* (2004) : pp. 163-182.

Bruce Croft. « IR Challenges and Language Modeling ». (2002) : .

Brunet-Manquat F. *Création d'analyseurs de dépendance par combinaison d'analyseurs syntaxiques*. Thèse de doctorat, Université Joseph Fourier. 2004.

Chevallet J. « X-IOTA Une plateforme distribuée ouverte pour l'expérimentation en Recherche d'Information ». *COnférence en Recherche Information et Applications CORIA'2004* (2004) : pp. 287-304.

Chevallet J. *Un Modèle Logique de Recherche d'Informations appliqué au formalisme des Graphes Conceptuels. Le prototype ELEN et son expérimentation sur un corpus de composants logiciels*. Thèse de doctorat, Université Joseph Fourier. 1992.

Chevallet J, Lim J & Radhouani S. « Using Ontology Dimensions and Negative Expansion to solve Precise Queries in CLEF Medical Task ». *CLEF Workhop, Working Notes Medical Image Track,* (2005) : .

Chow C & Liu C. « Approximating discrete probability distributions with dependence trees ». *IEEE Trans. Inf. Theory* (1968) 14: pp. 462-467.

Cui H, Kan M, Chua T. « Generic soft pattern models for definitional question answering ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (2005) : pp. 384-391.

Delbecque T, Jacquemart P & Zweigenbaum P. « Utilisation du réseau sémantique de l'UMLS pour la définition de types d'entités nommées médicales. ». *CORIA* (2005) : pp. 101-118.

Fagan J. « Automatic phrase indexing for document retrieval ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (1987) : pp. 91-101.

Fairthorne RA. « Content analysis, specification and Control ». *Annual Review of Information Science and Technology* (1969) 4: pp. 73-109.

Ferret O, Grau B, Hurault-Plantet M, Illouz G & Jacquemin C. « Utilisation des entités nommées et des variantes terminologiques dans un système de question-réponse ». *TALN'2001* (2001) : pp. 153-162.

Gao J, Nie J, Wu G & Cao G. « Dependence language model for information retrieval ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (2004) : pp. 170-177.

Gaussier É, Grefenstette G, Hull D & Roux C. « Recherche d'information en français et traitement automatique des langues ». *Traitement Automatique des Langues* (2000) 41: .

Genest D. *Extension du modèle des graphes conceptuels pour la recherche d'information*. Thèse de doctorat, université Montpellier II. 2000.

Genest D & Chein M. « A Content-search Information Retrieval Process Based on Conceptual Graphs ». *Knowledge And Information Systems* (2005) 8: pp. 292-309.

Gonzalo J, Verdejo F, Chugur I & Cigarran J. « Indexing with WordNet synsets can improve Text Retrieva ». *Proceedings of the COLING/ACL '98 Workshop on Usage of of WordNet for NLP* (1998) : pp. 38-44.

Haddad H & Chevallet J. « Utilisation des syntagmes nominaux pour la Recherche d'information ». *EGC'2003 Journées francophones d'Extraction et de Gestion des Connaissances, Atelier "Fouilles de données et recherche d'informations dans des bases de données multi-média semi-structurées"* (2003) : .

Hersh WR & Donohoe LC. « SAPHIRE International: A tool for cross-language information retrieval ». *Proceedings of the American Medical Informatics Association Annual Fall Symposium* (1998) : pp. 673-677.

Hiemstra D. « A Linguistically Motivated Probabilistic Model of Information Retrieval ». *European Conference on Digital Libraries* (1998) : pp. 569-584.

Ho B. *Vers une indexation structurée, basée sur des syntagmes nominaux. Impact sur un SRI en vietnamien et la RI multilingue.*. Thèse de doctorat, Université Joseph Fourier. 2004.

Huang Y, Lowe H & Hersh W. « A pilot study of contextual UMLS indexing to improve the precision of concept-based representation in XML-structured clinical radiology reports. ». *Proceedings of the conference of the American Medical Informatics Association* (2003) : pp. 580-587.

Kahane S. « Grammaires de dépendance formelles et théorie Sens-Texte, Tutoriel ». *Actes TALN 2001, vol. 2* (2001) : pp. 17-76.

Katz B & Lin J. « Selectively Using Relations to Improve Precision in Question Answering. ». *Proceedings of the EACL-2003 Workshop on Natural Language Processing for Question Answering* (2003) : .

Kefi-Khelif L. *Un modèle général de recherche d'information : Application à la recherche de documents techniques par des professionnels*. Thèse de doctorat, Université Joseph Fourier. 2006.

Kleinsorge R, Willis J & Cathcart G. « Unified Medical Language Unified Medical Language System System (UMLS) Basics ». (2006) : .

Lee C & Lee GG. « Probabilistic information retrieval model for a dependency structured indexing system ». *Information Processing and Management* (2005) 41: pp. 161-175.

Lee C, Lee CJ & Myung Gil. « Dependency Structure Applied to Language Modeling for Information Retrieval ». *ETRI Journal* (2006) 28: pp. 337-346.

Lin D. « Dependency-based evaluation of Minipar ». *Workshop on the Evaluation of Parsing Systems* (1998) : .

Lin D & Pantel P. « Discovery of inference rules for question-answering ». *Natural Language Engineering* (2001) 7: pp. 343-360.

Losee RMJ. « Term dependence: truncating the Bahadur Lazarsfeld expansion ». *Information Processing and Management* (1994) 30: pp. 293-303.

Maisonnasse L. « Vers l'exploitation d'analyse de dépendance en recherche d'information précise ». *INFORSID 2005* (2005) : pp. 505-520.

Maisonnasse L & Tambellini C. « Dépendances syntaxiques et méthodes de détection de passages pour une segmentation sur le locuteur et le thème, Défi fouille de textes : reconnaissance automatique des auteurs de discours ». *Revue des Nouvelles Technologies de l'Information - RNTI - E10* (2007) : pp. 85-104.

Maisonnasse L, Gaussier E & Chevallet J. « Revisiting the Dependence Language Model for Information Retrieval ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (2007) : pp. 695 - 696.

Maron ME & Kuhns JL. « On Relevance, Probabilistic Indexing and Information Retrieval ». *Journal of the ACM* (1960) 7: pp. 216-244.

Martinet J. *Un modèle vectoriel relationnel de recherche d'information adapté aux images*. Thèse de doctorat, Université Joseph Fourier. 2004.

Matsumura A, Takasu A & Adachi J. « The Effect of Information Retrieval Method Using Dependency Relationship Between Words ». *RIAO 2000* (2000) : .

Meghini C, Sebastiani F & Straccia U. « Mirlog: a logic for multimedia information retrieval ». *Information Retrieval: Advanced models for the representation and retrieval of information.* (1998) : .

Mel'čuk I. « Vers une linguistique Sens-Texte. ». *Leçon inaugurale, Collège de France, Chaire internationale* (1997) : .

Metzler D & Haas S. « The constituent object parser: syntactic structure matching for information retrieval ». *ACM Transactions on Information Systems* (1989) 7: pp. 296-316.

Mitra M, Buckley C, Singhal A & Cardie C. « An analysis of statistical and syntactic phrases ». *Proceedings of RIAO97, 5th International Conference "Recherche d'Information Assistee par Ordinateur"* (1997) : pp. 200-214.

Mugnier M & Chein M. « Représenter des connaissances et raisonner avec des graphes ». *Revue d'Intelligence Artificielle* (1996) 10: pp. 7-56.

Mulhem P, Leow WK & Lee YK. « Fuzzy Conceptual Graphs for Matching Images of Natural Scenes ». *In proceedings of IJCAI 01* (2001) : pp. 1397-1404.

Müller H, Deselaers T, Lehmann T, Clough P & Hersh W. « Overview of the ImageCLEFmed 2006 medical retrieval and annotation tasks ». *Evaluation of Multilingual and Multi-modal Information Retrieval -- Seventh Workshop of the Cross-Language Evaluation Forum, CLEF 2006* (2007) 4730: pp. 595-608.

Nadkarni P, Chen R & Brandt C. « UMLS concept indexing for production databases: a feasibility study ». *Journal of the American Medical Informatics Association : JAMIA*. (2001) : pp. 80-91.

Nallapati R & Allan J. « Capturing term dependencies using a language model based on sentence trees ». *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management* (2002) : pp. 383-390.

Nie J. *Un modèle logique général pour les Systèmes de Recherche d'Informations. Application au prototype RIME*. Thèse de doctorat, Université Joseph Fourier. 1990.

Ounis I & Pasca M. « Effective and Efficient Relational Query Processing using Conceptual Graphs ». *BCS Colloquium on Information Retrieval (IRSG'98)* (1998) : .

Peters C, Gonzalo J, Braschler M, Kluck M. « Comparative Evaluation of Multilingual Information Access Systems ». *4th Workshop of the Cross-Language Evaluation Forum, CLEF 2003* (2003) : .

Polguère A. « La théorie Sens-Texte ». *Dialangue* (1998) 8-9: pp. 9-30.

Ponte JM & Croft WB. « A language modeling approach to information retrieval ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (1998) : pp. 275-281.

Punyakanok V, Roth D & Yih W. « Mapping dependencies trees: An application to question answering ». *Proceedings of AI&Math 2004*. (2004) : .

Robertson SE & Walker S. « Some simple effective approximations to the 2–Poisson model for probabilistic weighted retrieval ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (1994) : pp. 232-241.

Robertson SE, Walker S, Jones S, Hancock-Beaulieu MM & Gatford M.. « Okapi at TREC-3 ». *TREC-3: text retrieval conference* (1995) : p. 109.

Roussey C. *Une méthode d'indexation sémantique adaptée aux corpus multilingues*. Thèse de doctorat, INSA de Lyon. 2001.

Salton G. *The SMART Retrieval System - Experiments on Automatic Document Processing*. Englewood Cliffs (Ed.). , 1971.

Sanderson M. « Word sense disambiguation and information retrieva ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (1994) : pp. 49-57.

Sebastiani F. « A probabilistic terminological logic for modelling information retrieval ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (1994) : pp. 122-130.

Smeaton A. « Using NLP or NLP Resources for Information Retrieval Tasks ». (1999) : pp. 99-111.

Sparck Jones K, Walker S & Robertson SE. « A probabilistic model of information retrieval: development and comparative experiments ». *Information Processing and Management* (2000) 36: p. 779--808.

Srikanth M & Srihari R. « Biterm Language Models for Document Retrieval ». *Proceedings of ACM-SIGIR* (2002) : pp. 425-426.

Srikanth M & Srihari R. « Exploiting syntactic structure of queries in a language modeling approach to IR ». *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management* (2003) : pp. 476-483.

Strzalkowski T, Carballo JP & Marinescu M. « Natural Language Information Retrieval: TREC-3 Report ». *Text REtrieval Conference* (1994) : pp. 39-54.

Sérasset G. « Fiche résumée du projet Imag PRISM Pour une Recherche d'Information Sémantique Multilingue. ». (2003) : .

Tapanainen P. *Parsing in two frameworks: finite-state and functional dependency grammar*. Thèse de doctorat, University of Helsinki. 1999.

Tesnière L. *Éléments de syntaxe structurale*. Klincksieck (Ed.). , 1959.

Vechtomova O, Karamuftuoglu M & Robertson SE. « On document relevance and lexical cohesion between query terms ». *Information Processing and Management* (2006) 42: pp. 1230-1247.

Vintar S, Buitelaar P, VolkSemantic M. « Relations in Concept-Based Cross-Language Medical Information Retrieval ». *In Proceedings of the ECML/PKDD Workshop on Adaptive Text Extraction and Mining (ATEM)* (2003) : .

Voorhees E. « Using WordNet to disambiguate word senses for text retrieval ». *Proceedings of ACM SIGIR conference on Research and development in information retrieval* (1993) : pp. 171-180.

Wallis P. « Information retrieval based on paraphrase ». *Proceedings of PACLING* (1993) : .

Zhai C & Lafferty J. « Model-based feedback in the language modeling approach to information retrieval ». *Proceedings of the CIKM '01 conference* (2001) : pp. 403-410.

Zhai C, Tong X, Milic-Frayling N & Evans DA. « Evaluation of syntactic phrase indexing - CLARIT NLP track report ». *The Fifth Text Retrieval Conference (TREC-5)* (1997) : .

Zhou W, Yu C, Smalheiser N, Torvik V & Hong J. « Knowledge-intensive Conceptual Retrieval and Passage Extraction of Biomedical Literature ». *Research and Development in Information Retrieval* (2007) : pp. 655-662.

Zipf GK. *Human Behavior and the principle of least effort*. Cambridge MAP (Ed.). , 1949.

Zou Q, Chu WW, Morioka C, Leazer GH & Kangarloo H. « IndexFinder: A Method of Extracting Key Concepts from Clinical Texts for Indexing ». *AMIA 2003 Annual Symposium on biomedical and health informatics* (2003) : pp. 763-767.

PUBLICATIONS

Revues nationales

Maisonnasse L & Tambellini C, « Dépendances syntaxiques et méthodes de détection de passages pour une segmentation sur le locuteur et le thème ». *Défi fouille de textes : reconnaissance automatique des auteurs de discours - Campagne DEFT'05 (TALN'05)* - Revue des Nouvelles Technologies de l'Information - RNTI (2007) : pp. 85-104.

Conférences internationales

Maisonnasse L, Chevallet JP & Berrut C, « Incomplete and Fuzzy Conceptual Graphs to Automatically Index Medical Reports ». *NLDB 07*, Paris (2007): pp. 240-251. (Taux d'acceptation : 28%)

Maisonnasse L, Gaussier E & Chevallet JP, « Revisiting the Dependence Language Model for Information Retrieval ». *poster SIGIR 2007*, (2007).

Conférences nationales

Maisonnasse L, Gaussier E & Chevallet JP, « Modélisation de relations dans l'approche modèle de langue en recherche d'information » Conférence *en Recherche Information et Applications CORIA'2008*, Trégastel France (2008), pp. 305-320.

Radhouani S, Maisonnasse L, Lim JH, Le D & Chevallet JP, « Une Indexation Conceptuelle pour un Filtrage par Dimensions, Expérimentation sur la base médicale ImageCLEFmed avec le méta thésaurus UMLS ». COnférence *en Recherche Information et Applications CORIA'2006*, Lyon France (2006) pp. 257-269.

Maisonnasse L, « Vers l'exploitation d'analyse de dépendance en recherche d'information précise ». *INFORSID 2005*, Grenoble, (2005) : pp. 505-520.

Maisonnasse L, « Validation syntaxique de relations sémantiques pour la RI ». *Rencontres Jeunes Chercheurs en Recherche d'Informations (RJCRI'07),* 2007.

Maisonnasse L, « Intégration de connaissances syntaxiques dans les modèles de langue pour la RI ». *Rencontres Jeunes Chercheurs en Recherche d'Informations* (2006).

Campagnes d'évaluations

a) **Internationales**

Maisonnasse L, Gaussier E, Chevallet JP, « Multiplying Concept Sources for Graph Modeling ». *Lecture Notes in Computer Science*, à paraître.

Maisonnasse L, Gaussier E, Chevallet JP, « Multiplying Concept Sources for Graph Modeling ». *Workshop CLEF 2007* , budapest (2007).

Maisonnasse L, Sérasset G & Chevallet JP, « Using the X-IOTA System in Mono- and Bilingual Experiments at CLEF 2005 ». *Lecture Notes in Computer Science* (2006) 4022 : pp. 69-78.

Maisonnasse L, Sérasset G & Chevallet JP, « Using Syntactic Dependency and Language Model X-IOTA IR System for CLIPS Mono & Bilingual Experiments in CLEF 2005 », *Workshop CLEF 2005*, Vienne, Autriche, (2005) pp 21-23.

b) **Nationales**

Khalis Z, Tambellini C & Maisonnasse L, « A chaque corpus son découpage et une segmentation pour tous ». *DEFT 06*, fribourg (2006).

Maisonnasse L & Tambellini C, « Dépendances syntaxiques et méthodes de détection de passages pour une segmentation sur le locuteur et le thème ». *DEFT 2005, TALN 2005 tome 2*, Dourdan , (2005) : pp. 155-164.

Rapports

Maisonnasse L, *Une Étude Préliminaire à l'Utilisation d'UNL en Recherche d'Information*, rapport de DEA, Groupe MRIM - CLIPS-IMAG, Juin, 2004.