Une analyse récursive constructive pour la recherche du sens du texte de spécialité

Marta Franova*, Yves Kodratoff*, Lise Fontaine**
* CNRS, LRI, Bât. 490, U. Paris-Sud, 91405 Orsay
mf, yk @lri.fr
** Cardiff School of English, Cardiff University, PO Box 94, Cardiff, CF10 3EU Wales, GB
FontaineL@Cardiff.ac.uk

Résumé. Cet article décrit une chaîne de traitement de textes en vue de la découverte des traces linguistiques au sein des textes dans le but de simuler une compréhension « humaine ». L'accent est mis sur l'aspect récursif des tâches composant cette chaîne. Par voie de conséquence, les variables sur lesquelles s'effectuent les appels récursifs doivent présenter la propriété d'être constructibles et d'assurer la terminaison des appels récursifs. Le respect de ces deux propriétés implique des choix dans la linguistique utilisée pour décrire les textes, et la nécessité d'une intervention experte dans la programmation du système, afin d'introduire les connaissances permettant de casser les boucles infinies de calcul. Notre présentation ne décrit qu'une petite partie des problèmes linguistiques, mais illustre l'ensemble des problèmes plus généraux d'une analyse en compréhension. L'utilisation simultanée de plusieurs média montre encore plus clairement la nécessité d'utiliser des modules interactifs qui s'appellent récursivement les uns les autres.

1. Introduction

C'est une sorte d'évidence que la compréhension des textes écrits est due à l'interaction de nombreux types de connaissances, et que les systèmes cherchant à rendre compréhensibles à la machine des textes rédigés par des humains doivent prendre en compte ce fait, même s'ils ne cherchent qu'à simuler la compréhension. Les systèmes de génération, bien que traitant du problème inverse de génération par une machine de texte compréhensible à un humain, sont obligés de rendre explicites ces interactions, et illustrent un choix précis relatif à la nature des modules nécessaires et de leurs interactions, comme le montre, par exemple, la structure du générateur GENYSIS de Cardiff University [Fawcett et Tucker, 1990].

Cette évidence est encore plus frappante quand il s'agit d'un système multimédia. Le fait de combiner les supports sous-entend que la 'compréhension' de l'un aide à la compréhension de l'autre. Ces interactions peuvent évidemment être directes, mais dès qu'une difficulté surgit dans la reconnaissance d'un des objets présents sur un des supports, alors ce problème doit être empilé et ne sera dépilé que lorsque l'autre support aura fourni une réponse satisfaisante. Cet article illustre seulement quelques-uns des problèmes soulevés par la compréhension des textes, mais la même structure récursive est sous-jacente dans toutes les approches multimédia.

Le but de cet article est de proposer une étude de ces interactions à la lumière d'une approche récursive à la construction d'un système de compréhension automatique de textes rédigés en langage naturel. En particulier, il est élémentaire de savoir que la récursion, par exemple celle de f(x) ← g(f(x-1), f(x-2)), nécessite une condition d'arrêt (ici, par exemple, f(0) = 0, et f(1) = 1) et une façon de construire le domaine des fonctions, de sorte que l'appel récursif soit défini, ici, une façon de calculer x-1 et x-2 à partir de x. Les propriétés générales de tels domaines, appelés 'domaines constructibles', ont été étudiées par Franova [Franova, 1995 ;1998 ; Franova et Kooli, 1998]. Nous discuterons les conséquences 'évidentes' de telles propriétés sur celles d'un système d'analyse récursive du langage naturel. Une autre propriété de la récursion est qu'il existe deux formes non triviales de récursion, la récursion primitive récursive et la récursion Ackermann-récursive. La première, de façon intuitive, prend en compte les effets d'autres calculs alors que la seconde prend en compte (récursivement) les effets des effets d'autres calculs. Nous traiterons dans un article ultérieur de ces deux formes possibles d'un système de compréhension de textes.

Nous construisons un analyseur de textes dans le but de reconnaître la présence de concepts (définis de façon non récursive!), intéressants pour un expert, au sein des textes. Cette reconnaissance se fait en repérant les traces linguistiques de ces concepts, ce qui exige une chaîne de traitement des textes, depuis leur acquisition jusqu'à la définition des traces linguistiques selon le schéma (simplifié, voir paragraphe 4) suivant :

acquisition → mise en forme ('nettoyage') → création du lexique → étiquetage grammatical → détermination des collocations pertinentes → résolution des coréférences → regroupement des collocations en 'traces de concepts'.

Chacune de ces étapes est effectuée par un module de traitement de textes qui pour faire sa part de travail, et comme nous l'avons déjà signalé, doit faire appel aux résultats d'autres modules. Il est donc important de s'assurer que ces interactions ne conduisent pas à des boucles infinies de calcul, d'où la nécessité de vérifier que chaque appel récursif a des conditions d'arrêt qui seront atteintes. C'est pourquoi les divers modules doivent dépendre de variables définies de façon très particulière, afin d'obtenir un système constructible. Ainsi, notre système, bien que sans prétention aucune à constituer un modèle de la compréhension humaine, illustre néanmoins une approche constructive (quelques fois même laborieuse) à la description de la compréhension, plutôt qu'une approche par 'flashs' de compréhension immédiate.

2. Un exemple

Donnons un exemple de traitement récursif, relatif à l'étiquetage grammatical d'un texte journalistique américain. Bien entendu, ce traitement dépend des ressources dont on dispose. Comme nous décrivons une approche fonctionnelle que nous voulons convergente, nous ne devons pas faire appel à des ressources dont le recueil est un problème encore plus difficile que le problème en cours. Par exemple, une ontologie universelle de tous les rôles possibles de tous les mots du vocabulaire permettrait (peut-être) de comprendre le contenu de tous les textes, mais la constitution d'une telle ontologie est encore une fiction. Inversement, certaines connaissances peuvent clairement être acquises à partir des textes mêmes que nous somme en train d'analyser et la constitution de ces ressources sera considérée comme possible dans ce qui suit.

L'exemple de la correction de l'étiquetage grammatical est particulièrement significatif dans le contexte des textes de spécialité. En effet, il existe d'excellent étiqueteurs grammaticaux, mais ils reposent sur un apprentissage effectué sur un corpus préalablement étiqueté par des humains. N'insistons pas sur les fautes que les humains peuvent faire, mais plutôt sur le fait que l'obtention d'un corpus étiqueté dans un domaine de spécialité est une opération extrêmement coûteuse. Par exemple, même en biologie moléculaire où de gros efforts universitaires sont en cours, nous n'avons pas connaissance d'un corpus étiqueté grammaticalement, de taille raisonnable, et qui soit à la disposition des chercheurs. Le problème d'un étiquetage grammatical correct reste donc d'actualité et la correction des fautes d'étiquetage grammatical d'un corpus de spécialité reste d'autant plus d'actualité.

Nous supposons toujours que le texte a déjà reçu un certain traitement et, ici, supposons que ses mots soient déjà étiquetés. Nous en sommes, disons, à la nième étape de récursion, et nous voulons, au cours de l'étape n d'étiquetage, corriger les éventuelles fautes de l'étiquetage réalisé à l'étape n-1. Notons que cette nième étape est encore loin de toute sémantique ou compréhension du texte qui, en tout état de cause, ne peut pas être obtenue avant la fin du processus, comme pour toute structure calculatoire récursive. Dans certains cas, cette correction pourra être effectuée directement, dans d'autres cas, il faudra empiler les problèmes afin de les résoudre dans une étape ultérieure. Dans ce cas, le problème est de savoir quelles erreurs doivent être obligatoirement éliminées car elles ne pourront pas être corrigées plus tard, et lesquelles peuvent être conservées. Très souvent, on a le choix entre plusieurs erreurs et la 'vérité'. Mais cette dernière est tellement coûteuse en temps qu'il vaut mieux admettre une erreur qui sera corrigée sans problème à une étape ultérieure.

Considérons un premier exemple, de confusion entre un verbe à la troisième personne du singulier et un nom pluriel.

Supposons que nous ayons obtenu l'étiquetage suivant :

> 'Ahimbisibwe/NP also/RB said/VBD he/PP saw/VBD a/DT man/NN carrying/VBG a/DT hammer/NN and/CC nails/VBZ early/JJ March_17/NP ./.' (Associated Press Worldstream News Service, 4 mars 2000)

Nous utilisons le vocabulaire du 'Penn Treebank', voir [Santorini, 1990], et 'VBZ' signifie *verbe à l'indicatif présent , 3ème personne*. La faute 'nails/VBZ' ne peut pas être repérée par le fait qu'on ne cloue pas les mois de l'année pour deux raisons. D'une part, dans notre exemple, une autre faute (de nettoyage – due à l'utilisation de WordNet) nous conduit à voir March_17 comme une locution figée et non comme une date, qu'on peut étiqueter soit comme un nom ('NN') soit comme un nom propre ('NP'), et on peut clouer des objets et même des individus alors qu'on ne peut pas clouer le temps. Deuxièmement, la connaissance de tout ce qu'on ne peut pas clouer est en soi une connaissance appartenant à une ontologie universelle qui n'est pas à notre disposition.

Un humain confronté à cet étiquetage grammatical repère immédiatement que « a man carrying a hammer and nails» contient un 'and' qui rend cette phrase incompréhensible si on voit *nails* comme un verbe (c'est-à-dire que hammer et nails jouent le même rôle dans la phrase). Cette faculté dépend soit de la pragmatique de l'usage des marteaux, soit de la compréhension globale de la phrase, et nous n'avons pas encore atteint cette étape.

Un humain repère aussi immédiatement que la phrase contenant nails/VBZ ne respecte pas la concordance des temps. Le problème est de savoir qui doit l'emporter : les formes au passé de said/VBD et saw/VBD, ou la forme au présent de nails/VBZ ?

Notre solution consiste à remarquer que, du fait de la distribution du présent de l'indicatif en anglais, le style journalistique fait rarement usage de la troisième personne du

singulier excepté pour deux classes particulières de verbes, les verbes relationnels [Halliday, 1994] (p. ex. remains, seems, appears, contains, looks like, etc.) et les verbes mentaux [Halliday, 1994]. (p. ex. thinks, knows, believes, etc.). Il est tout à fait facile de vérifier cette propriété sur les textes réels, et tout à fait possible de faire une liste de tels verbes, tels qu'ils sont utilisés dans le style journalistique, à partir d'articles de journaux. On vérifie aussi que, lorsque le journaliste décrit un processus matériel avec un verbe à la troisième personne du singulier, alors le texte est rendu compréhensible au lecteur par un usage non ambigu de la forme «sujet, verbe» et le sujet n'est pas lui-même ambigu, dans la mesure où nous supposons que le rédacteur fait un effort de clarté pour ses lecteurs. On remarque que cette liste peut différer pour les textes techniques, par exemple. Notre approche ne prétend pas à l'universalité mais dépend des styles utilisés. En d'autres termes, nous supposons que nous disposons d'un moyen de reconnaître si un texte appartient à un style connu avant de commencer à l'analyser et que nous avons une ontologie des styles connus. Cette ontologie n'est donc pas nécessairement universelle est n'est donc pas une fiction. Des systèmes automatiques comme les n-grammes ou les SVM peuvent résoudre ce problème avec une grande précision.

C'est pourquoi nous suggérons d'utiliser ici cette propriété : 'nails/VBZ' est un verbe ni relationnel ni mental et il est précédé d'une séquence d'étiquettes '/NN /CC' qui le sépare de son hypothétique sujet.

Nous proposons donc la formulation récursive suivante au problème des noms terminés par un 's' et pouvant être un verbe, et étiquetés VBZ : si le verbe n'appartient pas à la liste des verbes relationnels ou mentaux, alors mettre le problème dans la pile jusqu'au stade de la détermination des sujets et des compléments. A ce stade, si le VBZ n'a pas de sujet, alors le réétiqueter nom pluriel ('NNS').

En passant, on remarque que nous avons résolu de façon récursive mais simple un problème de compréhension qui pourrait sembler faire appel même à la pragmatique : pourquoi la phrase « a man carrying a hammer and nails **/VBZ** early/JJ March_17/NP» n'a pas de sens. C'est parce 'nails' n'a pas de sujet évident et non pas parce que l'opération de 'clouer un 17 mars précoce' n'a pas d'instance ancrée dans la monde réel.

Cet exemple est central à notre propos car il illustre comment on définit le domaine constructible des systèmes récursifs de compréhension du langage naturel écrit.

Donnons un autre exemple, illustrant comment plusieurs fautes imbriquées peuvent compliquer le problème. Nous avons rencontré l'analyse suivante :

> 'flopping/VBG fingers/VBZ input/NN numbers/NNS and/CC output/NN commands/NNS ./. (Xinhua News Service 21 novembre 1999).

Commençons par l'analyse du premier mot : flopping/VBG.

C'est un mot en «ing» qui peut être un participe présent (comme le lexique nous l'indique). D'après le vocabulaire de 'Penn Treebank', nous savons donc que 'flopping' peut faire fonction dans la phrase soit de participe présent (étiqueté 'VBG'), soit un adjectif, étiqueté, 'JJ', *premodification by –ing* selon la terminologie de [Quirk *et al.*, 1985]. Par contre, Penn Treebank ne fait pas la différence entre les deux usages possibles, fini ou non fini, du verbe.

'flopping' est reconnu comme un verbe ni relationnel ni mental et donc soit il doit introduire une clause soit c'est un JJ. L'appel récursif à la recherche d'un complément renvoie la connaissance que l'étiquette VBG est impossible (puisque 'fingers' a été étiqueté VBZ). On voit donc ici que l'analyse serait complètement différente (nous ne la ferons pas) si 'fingers' avait été dès le départ étiqueté correctement NNS.

L'analyse du mot fingers/VBZ conduit à une impasse puisque 'to finger' est un verbe qui est ni relationnel ni mental, et à l'étape de découverte du sujet, le système revient en disant que fingers ne peut pas être un VBZ. L'analyse conclut finalement à flopping/JJ fingers/NNS (sans faire appel à la pragmatique qui nous dit qu'un 'affalement ou une souplesse' ne peut pas 'toucher').

3. Le texte de spécialité et la récursion

L'exemple ci-dessus illustre une position plus systématique de notre part. Dans tous les cas, nous pensons que l'appel à des connaissances pragmatiques doit être vu comme un dernier recours, du fait de sa complexité pour des domaines non caricaturaux.

Plus généralement, le texte de spécialité émane d'un auteur qui cherche à éviter les ambiguïtés du langage, au lieu de les favoriser comme cela serait possible dans un domaine littéraire. Ainsi, notre approche récursive ne se pose pas le problème de l'analyse d'un langage hypothétique contenant toutes les phrases grammaticalement correctes de la langue, mais celui des phrases non ambiguës qui 'devraient' pouvoir être correctement annotées de façon automatique.

Cette hypothèse nous permet d'envisager une implémentation récursive d'un analyseur en compréhension. Notons encore que nous ne prétendons pas résoudre automatiquement le problème dit 'du passage de la syntaxe à la sémantique'. Nous verrons plus loin, voir aussi [Kodratoff, 2005], comment nous abordons ce problème avec l'aide d'un expert du domaine. Par contre, nous essayons en effet de résoudre le problème des annotations à ajouter au texte de sorte que l'expert puisse y adjoindre sans erreur la sémantique qu'il désire voir apparaître.

4. Une implémentation récursive

Afin d'arriver à aborder le problème de la compréhension du texte, nous introduisons une chaîne de traitement dont nous pensons qu'elle autorise une programmation récursive du fait de l'ordonnancement de ses modules. Autrement dit, la chaîne n'est pas linéaire puisque chacun de ses modules peut faire appel à un autre module. Néanmoins, l'ordre dans lequel les modules construisent la compréhension est tel qu'aucun appel récursif ne doit conduire à un problème plus complexe que le problème de départ. Chaque module de rang p lance un appel récursif à un module de rang p+k seulement si ce problème est bien posé (au rang p) pour le rang p+k.

Dans la mesure où ce logiciel est en cours d'évolution, nous devons avouer qu'il existe des cas où la solution utilisée est triviale : si le problème formulé au rang p est mal posé pour le rang p+k alors on abandonne le problème et on conserve une erreur possible au rang p. Bien évidemment, notre effort consiste à éliminer le plus possible ces solutions triviales.

Dans notre présentation, nous utiliserons souvent le fait que nous disposons de plusieurs textes traitant du même sujet afin d'aller chercher dans les textes voisins une information nécessaire à la compréhension du texte étudié, par exemple pour construire une ontologie relative aux objets, aux événements ou aux individus dont le texte parle. Deux problèmes différents sont à traiter pour recueillir puis utiliser les textes voisins.

La constitution de ce sous-corpus de textes voisins peut bien entendu être fortement récursive en ce sens que selon l'étape d'analyse où l'on se trouve sur un texte, la constitution du sous-corpus peut se modifier. Nous n'avons pas abordé ce problème.

Une utilisation récursive des textes voisins est possible, mais nous avons considéré qu'il fallait limiter autant que possible ces appels récursifs qui peuvent être très coûteux en temps, et nous avons donc préféré effectuer une recherche systématique sur les textes voisins afin d'en tirer les ontologies nécessaires à la compréhension d'un des textes.

Dans la présentation de la chaîne de traitement qui suit, nous ne parlons pas de la phase de recueil automatisé que nous avons très peu étudiée et qui constitue un vaste objet de recherche, comprenant tous les efforts de constitution de corpus.

4.1. Nettoyage

(comprenant la délimitation des phrases et des paragraphes, la recherche des abréviations et acronymes, et des locutions figées).

La recherche des paragraphes peut apparaître surprenante et sans intérêt pour le traitement linguistique. En fait, nous nous sommes aperçus de profondes différences dans la recherche des coréférences à l'intérieur et à l'extérieur d'un même paragraphe. Cette information de nature visuelle n'étant plus disponible (en général) après nettoyage, il est nécessaire de la recueillir très tôt.

La phase de nettoyage peut détruire d'autres informations potentiellement utiles comme , par exemple, la présence de majuscules au début des mots ou les références à d'autres textes.

Selon l'usage que l'expert du domaine veut faire du texte étudié, les options de nettoyage peuvent être très différentes. Dans les textes de journalisme que nous prenons en exemple ici, le repérage des noms de personnes est capital pour la compréhension du texte, donc, par exemple, il est dangereux de confondre (Teddy) Rose et 'rose *nom* 'NN' ou *verbe au passé* 'VBD'). Inversement, dans des textes de spécialité scientifique, il est peut être inutile de créer des logiciels complexes pour éviter des confusions. Par exemple, un géologue nommé Mr. Rose ne crée pas de confusion, il faudrait qu'il porte le nom d'un terme technique, comme Mr. Layer, mais ce cas sera très rare.

Plus généralement, on s'aperçoit que même la phase 'triviale' de nettoyage comporte des étapes que seul un expert du domaine peut traiter. Lorsque la compréhension du texte repose sur ce type de connaissance, l'expert s'aperçoit de l'erreur durant des étapes ultérieures, par exemple en rencontrant des collocations 'absurdes' (pour l'expert). La récursion que nous utilisons alors devient en apparence triviale d'un point de vue théorique : nous fournissons simplement à l'expert le moyen de corriger cette erreur dans le module . Remarquons cependant d'une part qu'un logiciel de nettoyage comporte des milliers de règles et leur conception de sorte que l'expert puisse corriger l'une d'elle sans introduire d'erreurs dans les autres est une tâche délicate. D'autre part, cela nous permet de simplifier l'analyse linguistique qui devient spécifique à un domaine particulier.

4.2 Lexique

(comprenant les étiquettes grammaticales possibles des mots et la recherche des entités nommées).

Nous utilisons un lexique déduit en partie de WordNet (http://www.cogsci.princeton.edu/~wn/).

Les étiquettes grammaticales possibles sont issues de grammaires en ligne. Ceci introduit de nombreuses possibilités peu probables. Par exemple, la chaîne de caractères 'have' peut être un nom ou un verbe, mais les occurrences du nom, en dehors d'une expression figée, n'existent pas, surtout dans des textes de spécialité. Les procédures mises en œuvre pour traiter ce problème n'étant pas récursives, nous ne les décrirons pas ici.

L'utilisation de cette ressource pose le problème du temps requis pour parcourir le lexique qui comporte plusieurs centaines de milliers de mots. Les procédures que nous avons mis en place évitent autant que possible de faire appel au lexique.

Par contre, nous avons été obligés d'introduire des appels récursifs pour l'étiquetage des noms propres. Nous disposons de listes de noms propres, de prénoms, de noms d'institution et de noms de pays. Nous avons dû introduire une différence d'étiquetage entre les noms propres et les noms de compagnie et de pays. Par exemple, pour résoudre une anaphore, considérons un 'his', qui réfère donc à un humain. Si un nom de pays se trouve entre ce 'his' et la dernière personne citée, notre système, du fait qu'il ne fait pas appel au dictionnaire, doit trouver dans le texte une étiquette lui indiquant s'il s'agit d'un humain ou d'un objet.

Les étiquettes elles-mêmes peuvent être erronées. Du fait de l'absence de ce nom de club sportif dans nos listes, nous avons systématiquement étiqueté Manchester/NP united/VBD. Si 'United' est écrit avec une majuscule, on pourrait aussi obtenir Manchester/NP United/NP. Nous n'obtenons donc jamais la forme désirée en locution figée : Manchester_United/NP. Si on adopte comme solution de relier en locutions figées tous les noms propres qui se suivent, alors on va trouver des President_Clinton et des Senator_Daniel_Patrick_Moynihan dont la fonction sera perdue. En conséquence, les références à 'Mr. President' ou 'the senator' seront perdues. La solution simple à ce problème consiste à avoir des listes de fonctions. Reste encore le problème de tous les cas inconnus ou ambigus (comme 'Prince'), qui demanderont une recherche récursive de fonctions d'un individu dans les textes proches de celui étudié pour savoir si l'on parle d'une fonction ou d'un individu.

4.3 Étiquetage grammatical

(en utilisant divers niveaux de contextualité: le mot, la phrase, les phrases environnantes, le contexte 'social')

Comme nous l'avons signalé plus haut, nous utilisons l'étiquetage de Penn Treebank, ce qui implique des choix implicites sur ce qui est important (et possible) dans la première étape d'étiquetage vers une compréhension du texte, c'est-à-dire, pour nous, vers un étiquetage de groupes de mots par un nom de concept. Ces choix sont en apparence arbitraires en ce sens qu'ils ne sont pas du tout universels. Par exemple, l'autre logiciel de base que nous avons utilisé, LinkParser, effectue une analyse syntaxique profonde et utilise des étiquettes différentes, bien sûr par leur nom, mais aussi par leur sémantique, comme nous avons pu le constater en faisant le lien entre ces deux systèmes d'étiquetage.

Dans le contexte de cet article, nous pouvons formuler ce choix en disant que nous avons supposé que l'étiquetage de Penn Treebank constituait un choix de variables appartenant à un domaine constructible. Nous n'avons pas encore vérifié l'exactitude complète de cette dernière assertion, mais nous avons en effet vérifié que certains choix semblant très arbitraires sont de fait nécessaires à la constructibilité du domaine des appels récursifs. Par exemple, Penn Treebank confond en une seule étiquette JJ les adjectifs, les numéraux, les participes présents et participes passés en position de prémodifieurs. Cette simplification grammaticale, bien que difficile à réaliser sans erreur, est capitale pour repérer les verbes en position d'auxiliaires, et ensuite les formes passives, et ensuite encore les clauses matérielles. Dans ces conditions, et dans le cadre d'appels récursifs, l'effort de reconnaissance des JJ ne doit pas porter sur l'analyse fine des participes ambigus, mais sur l'usage ou non d'une forme passive. En d'autres termes, et de façon plus générale, la règle d'attribution d'une étiquette ne dépend pas seulement de la nature grammaticale exacte de cette étiquette (qui a été définie par des grammairiens ayant déjà une compréhension globale de la phrase) mais aussi de l'usage ultérieur de cette étiquette afin de préparer la liaison entre programme appelant et programme appelé dans la chaîne de traitement linguistique.

Les informations nécessaires au choix d'une étiquette correcte peuvent dépendre de nombreux contextes de différents niveaux. En section 2, nous avons illustré une règle d'étiquetage contextuel qui dépend à la fois du contexte dans la phrase (présence non ambiguë d'un 'sujet' et d'un 'objet') et du contexte social (règle valable seulement pour certains genres, ici le genre journalistique).

Jusqu'à présent nous avons effectué un choix de règles à utiliser pour définir le contexte utile dont nous reconnaissons qu'il est encore très expérimental. Une théorisation de ces choix est extrêmement délicate et constitue une linguistique de spécialité dont les liens avec la linguistique générale doivent être analysés avec minutie. Nous n'avons pas encore entrepris ce travail de façon systématique.

En tout état de cause, les exemples de la section 2 illustrent une idée centrale à cet article : ils décrivent les variables sur lesquelles s'effectuent la récursion, qui donc déterminent si le domaine sur lequel s'effectuent nos appels récursifs est ou non un domaine constructible.

4.4 Recherche des collocations 'nominales'

(collocations appartenant à la partie non verbale de la clause)

Dans le cadre d'un texte de spécialité, nous définissons la notion de collocation par son sens pour un expert. Même une locution figée, comme 'data mining', est une collocation intéressante en informatique, en ceci qu'elle définit un domaine de recherche précis pour un expert en informatique.

Notons que cette recherche doit avoir lieu avant la constitution des groupes verbaux si on désire travailler sur un domaine constructible. En effet, la recherche simultanée des groupes nominaux et des groupes verbaux constitue la base d'une analyse syntaxique profonde de la phrase, un processus connu pour son extrême lenteur, comme nous l'avons expérimenté lors de notre utilisation de LinkParser [Sleator et Temperley, 1991].

La méthode que nous utilisons a déjà été publiée plusieurs fois, voir par exemple [Roche, 2003] et se fonde sur une construction récursive des termes. Dans la publication référencée, nous parlons de construction itérative car nous utilisons une forme simplifiée de

la récurrence où la pile des problèmes est remplacée par un ‘accumulateur’ où sont stockés les résultats intermédiaires.

4.5 Résolution des coréférences

(incluant un recueil automatisé d'ontologies spécifiques au contexte 'social')

L’importance du contexte social dans la compréhension des textes a été souvent soulignée par Halliday, par exemple dans [Halliday et Hasan, 1985; Halliday, 1994]. Dans notre pratique, nous considérons comme un contexte social l’ensemble des textes proches. Par exemple, les textes relatifs à Mme Clinton pendant la période où Mr. Clinton était président, parlent d’elle comme la ‘first_lady/NN’, une locution figée fournie par Wordnet. Ainsi, on pourrait croire que, relativement à Mme Clinton, les textes parlant d’elle en tant que candidate sénateur de l’état de New York, appartiennent à un contexte social différent. Le contexte social réel fait que les deux titres ‘first_lady et ‘candidate’ se trouvent dans les mêmes textes, comme l’illustre bien cette phrase :

> 'a spokeswoman for the First Lady's New York Senate campaign on Friday said President Clinton will join his wife Tuesday to cast his first vote in New York State.'
> (Associated Press Worldstream News Service, 8 septembre 2000).

Considérons que pour résoudre les coréférences (en particulier, la difficile séquence anaphorique ‘his’ ← President Clinton’s, et President Clinton’s wife ← first_lady) il nous faut d’abord construire une ontologie des relations familiales – bien entendu sans connaître encore la solution des coréférences.

Pour construire cette ontologie, et si on n’observe que cette phrase seule, et si le nom de ‘Clinton’ ainsi que le titre de ‘first_lady’ ont déjà été associés à Mme Clinton dans d’autres phrases de ce texte, il est naturel de lui associer maintenant aussi la fonction de ‘President’, surtout pendant cette étape préalable à la résolution des coréférences. En fait, ou bien on suppose que les relations de Mr. et Mme Clinton sont déjà connues, et le problème devient trivial dans ce cas, mais incroyablement complexe en général, puisqu’il nous faut alors inclure les relations familiales de tous les chefs d’état du monde, par exemple celles du président Allende et de Mlle Isabel Allende, sans parler des parents, amis et alliés de toutes les personnes ayant défrayé la chronique. Dans ces conditions, on voit bien qu’un appel à une procédure automatique de construction de taxonomie à partir des textes est absolument indispensable. L’hypothèse de non ambiguïté que nous avons déjà formulée nous permet d’utiliser une règle interdisant à deux individus portant le même nom et ayant une relation familiale avérée d’exercer la même fonction dans le même contexte social.

4.6 Recherche des collocations 'verbales'

(collocations contenant la forme verbale de la clause. Ceci inclut une lemmatisation)

Afin d’obtenir un domaine constructible, nous avons choisi d’utiliser une approximation à l’analyse syntaxique, et même à l’analyse superficielle. Nous déterminons d’abord les formes pronominales, ce qui est tout à fait possible en utilisant seulement l’étiquetage grammatical, et nous associons un verbe à ce qu’on appelle d’habitude un ‘sujet’ ou un ‘complément’ comme étant le premier objet ou individu avant (resp. après pour les pronominaux) ou après (resp. avant pour les pronominaux). C’est pourquoi nous parlons de

collocations ou de termes verbe-nom et nom-verbe plutôt que de sujets et d'objets du verbe. Bien entendu cette règle de base peut être complexifiée pour prendre en compte les juxtapositions, les subordonnées et les coordinations. Mais cette complexification ne peut aller très loin sans retomber sur les problèmes de l'analyse lexicale profonde.

Comme ces termes seront la base des traces linguistiques de concepts, une lemmatisation peut maintenant être effectuée pour éviter la multiplication des formes terminologiques.

La solution que nous apportons à ce problème illustre très bien la nature récursive de notre approche. En effet, nous abandonnons la recherche de ces collocations dissimulées dans les phrases complexes, et nous allons donc perdre certaines collocations. Cependant, on peut argumenter du fait que seules deux possibilités se présentent. Ou bien la collocation 'perdue' est sans intérêt pour le sens des textes étudiés (pour reprendre le vocabulaire de Halliday, cette collocation est inutile dans le contexte social de la phrase), et alors la perdre n'a pas d'importance. Ou bien cette collocation est importante, et elle apparaîtra ailleurs, au sein d'autres phrases de ce même contexte social. Nous la classerons alors comme une collocation importante et le problème posé par la phrase où elle a été perdue devient facile à résoudre, c'est celui de la reconnaissance d'une collocation connue, dissimulée dans la phrase complexe. L'erreur que nous avons faite à la phase de reconnaissance devient très facile à corriger dans une phase ultérieure, il est donc inutile de s'acharner à corriger l'erreur de départ : la constructibilité du domaine nous autorise à déléguer un problème facile à résoudre plus tard.

On notera aussi que la phase de détermination des coréférences doit impérativement prendre place avant la recherche de ces collocations verbales. En effet, on réfère souvent aux individus, objets ou phénomènes soumis à une action par un pronom personnel, ou par un hyperonyme (p. ex. 'this murder') ce qui détruit la collocation explicite dans la phrase.

4.7 Acquisition de concepts à partir de textes

(recueil de connaissances expertes complété par un processus appelé: *induction extensionnelle*)

L'acquisition de concepts se fait de deux manières [Fontaine et Kodratoff, 2002].

Premièrement, une interface homme-machine, déjà décrite dans [Fontaine et Kodratoff, 2002; Kodratoff, 2005] est mise en place, afin de recueillir les concepts et les traces linguistiques de concepts. Cette interface n'est récursive que trivialement en ce sens que c'est le travail de l'expert de la définir. Par exemple, si l'expert remarque une erreur dans une entité nommée alors il doit modifier nettoyage et lexique. Un exemple frappant de ce type d'erreur est celle que nous avons commise en analysant des textes journalistiques appartenant à la compétition TREC *Novelty*. En définissant les collocations, nous avons rejeté, par étourderie, comme insignifiante la collocation 'first-woman' ou 'first-female'. C'est seulement en étudiant les textes relatifs à la première femme commandant une mission de la navette spatiale que nous nous sommes aperçus que seule la reconnaissance de ces collocations permettait de distinguer les actions de cette personne de celles effectuées par d'autres femmes astronautes[1].

[1] Cette erreur nous a considérablement gêné dans l'expression des concepts décrivant le comportement de cette personne aux commandes de sa navette.

Deuxièmement, l'expert du domaine ayant fourni un nombre 'suffisant' (cette notion n'est pas absolue et se définit par des essais-erreurs d'utilisation de la procédure inductive décrite ci-dessous) d'exemples de traces linguistiques, une procédure inductive automatique peut se mettre en place. Appelons les traces dont on part, les noyaux de traces linguistiques de description des concepts. Cette procédure a été déjà décrite dans [Kodratoff, 2004; 2005] et son fonctionnement est non trivialement récursif. En effet, l'acquisition de nouvelles traces de concepts est effectuée de façon d'abord itérative puisque les nouveaux concepts acquis automatiquement à l'itération k peuvent servir de nouveaux noyaux à l'itération k+1. Cependant, cette procédure peut conduire à l'introduction de contradictions : une trace donnée peut devenir représentative de plusieurs concepts différents. Pour détecter ces contradictions et pour éviter une divergence dans la phase itérative de l'induction, la procédure itérative doit faire appel récursivement à une procédure de décision qui élimine les contradictions à mesure qu'elles apparaissent.

4.8 Applications

L'application de la connaissance ainsi acquise n'est pas toujours récursive, mais nous voulons signaler ici l'importance de la reconnaissance des concepts dans un texte. La recherche par mots-clé, si décevante, devient évidemment très riche quand on recherche des concepts-clé indicateurs du sens du texte.

Une application récursive évidente est l'utilisation des concepts déjà repérés pour diminuer le taux d'erreur des étapes que nous venons de décrire. Cependant, nous n'avons pas encore implémenté ces appels récursifs de la procédure globale à elle-même.

5. Conclusions

Nous avons décrit une chaîne récursive de traitements linguistiques dont le rôle est double : assurer le contrôle permanent d'un expert du domaine sur chaque étape de la chaîne, et assurer que le domaine de définition des diverses variables du programme soit constructible.

Cette approche est une instance de construction d'un système complexe décrit par Franova [Franova, 2004]. Il obéit donc à la règle qui veut que le système ne soit vraiment convaincant que lorsqu'il est achevé. En fait, nous n'avons achevé que la première étape de cette construction. Notre application aux tâches de la compétition TREC-*Novelty* (http://trec.nist.gov) nous a montré que, même en l'absence d'un réel expert du domaine et même pour un temps de travail trop court, notre approche conduit à de très bons résultats en précision (les concepts définis même par le 'non-expert' sont la plupart du temps exacts)

Signalons toutefois que sur ce thème particulier et seulement relativement à la pertinence des phrases (ce qui constitue seulement 1/100ème des tâches à effectuer !), nous obtenons le meilleur résultat des 60 concurrents de la tâche 1 de la compétition TREC *Novelty* 2004 (publication en cours pour 2005). Ce dernier commentaire a pour but de montrer en passant que, pour théorique qu'elle soit, notre approche est susceptible d'application sur de grandes quantités de textes réels.

mais à des résultats catastrophiques en rappel. En effet, les concepts définis par un **non**-expert ne couvrent, en général, que très imparfaitement le domaine.

Ceci constitue cependant une toute première validation de notre approche en ce sens qu'elle a été au moins effectivement appliquée sur un problème réel, et que les résultats sont loin d'être décevants. Notre système étant conçu pour un véritable expert travaillant assez longuement (et non pas dans les conditions tendues d'une compétition), nous allons maintenant essayer de l'appliquer à un domaine remplissant ces conditions.

Remerciements

Le lexique et le nettoyage ont été programmés par Thomas Heitz; un langage pour corriger l'étiquetage grammatical ('CorTag') a été écrit par Jérôme Azé; l'interface d'étiquetage ('ETIQ') est due à Ahmed Amrani; le travail sur les collocations a été effectué par Mathieu Roche, Thomas Heitz en a écrit l'interface ('EXIT'); le logiciel ACT d'acquisition de traces de concepts est dû à Jérôme Maloberti et Ahmed Amrani. Le logiciel nécessaire à la phase inductive de construction des traces de concepts a été réalisé par Jérôme Azé.

Références

[Fawcett et Tucker, 1990] Fawcett R. P., Tucker G. H., Demonstration of GENESYS : a very large, semantically based systemic functional grammar, Proc. 13rd International Conference on Computational Linguistics (COLING-90), 1:47-49, 1990.

[Fontaine et Kodratoff, 2002] Fontaine, L. Kodratoff Y. La notion de 'concept' dans les textes spécialisés: une étude comparative entre la progression thématique et la texture des concepts , *ASp* 37-38:59-83, 2002.

[Franova; 1995] Franova M. A Theory of Constructible Domains - a formalization of inductively defined systems of objects for a user-independent automation of inductive theorem proving, Part I; Rapport de Recherche No.970, L.R.I., Université de Paris-Sud, Mai 1995.

[Franova et Kooli, 1998]. Franova M., M. Kooli: Theory of Constructible Domains for Robotics : Why? in: J. Mira, A. Pasqual del Pobil, M. Ali: *Methodology and Tools in Knowledge-Based Systems*; IEA-98-AIE, vol. I, LNAI 1415, Springer, 1998, 37-46.

[Franova, 1998]. Franova M., Reformulation and Formulation of Definitions: Why and How? -- Saying "Good-bye" to the notion of well-founded relation ; Report no. 43, Reports of the Austrian Society for Cybernetic Studies, published in March 1998. ISBN number: 3 85206 136 9.

[Franova, 2004] Franova M., Systèmes descarto-ackermanno-filkornisés: Définition et Applications; Rapport de Recherche No.1384, L.R.I., Université de Paris-Sud, Orsay, France, Mars 2004.

[Halliday et Hasan, 1985] Halliday, M.A.K. and R.Hasan, *Language, Context and Text : Aspects of Language in a Social-semiotic perspective*, Oxford:Oxford University Press, 1985.

[Halliday, 1994] Halliday, M.A.K.,. *An introduction to functional grammar*, 2nd ed. London:Edward Arnold, 1994.

[Kodratoff, 2004] Kodratoff Y. Induction extensionnelle: définition et application à l'acquisition de concepts à partir de textes, *RNTI*, 2: 247-252, 2004.

[Kodratoff; 2005] Kodratoff Y. Genre Specific Text Mining and Extensional Inductive Concept Recognition: A Pseudo-Cognitive Approach, in *Handbook of Categorization in Cognitive Science*, C. Lefebvre & H. Cohen (Eds.) Elsevier (2005).

[Quirk *et al.*, 1985]. Quirk R., Greenbaum S., Leech G., Svartvik J., *A comprehensive grammar of the English language*, Longman, London 1985.

[Roche, 2003] Roche M. Extraction paramétrée de la terminologie du domaine, RSTI RIA-ECA, 17:295-306.

[Santorini, 1990] Santorini B. Part-of-Speech Tagging Guidelines for the Penn Treebank Project, http://www.cis.upenn.edu/~treebank/home.html, 1990.

[Sleator et Temperley, 1991] Sleator D., Temperley D. Parsing English with a Link Grammar, Carnegie Mellon University Computer Science technical report CMU-CS-91-196, 1991.

Summary

In this paper we describe a series of text processing whose goal is to discover the linguistic « traces » (highly complex data) in texts which lead to simulating human understanding. We are focussing on the recursive nature of the interactions of the modules in the series. The variables involved in the recursive call must have the property of being constructible and of being able to ensure termination of the recursive call. Having to respect these two conditions imposes certain choices on the linguistic treatments required and it means that a human expert has to intervene in the processing system so that relevant knowledge can be introduced that will prevent infinite cycles. Our paper only describes a small part of the linguistic problems involved in the series, but it nevertheless illustrates the more general set of problems in dealing with this type of natural language understanding. With complex data such as the units of text comprehension, the need for simultaneous choice making among several media in the series demonstrates very clearly the need for interdependent modules that are mutually recursively called.

RNTI - E -