Chapitre VII

RÉSOLUTION DE LA RÉFÉRENCE : UN MODÈLE REPRÉSENTATIONNEL ET SON IMPLÉMENTATION

VII.A Introduction

Nous avons vu au Chapitre V un certain nombre d'approches du traitement de la référence, aussi bien théoriques qu'implémentées et évaluées. Parmi les modèles théoriques, nous avons mis en évidence l'intérêt des théories fondées sur la prise en compte du contexte et des intentions des locuteurs, et nous avons décrit le modèle d'inspiration pragmatique élaboré dans le cadre du projet CERVICAL. Au Chapitre VI, nous avons proposé un modèle d'utilisation de la référence par un agent, qui s'intégrera à terme dans une simulation conforme à la « pragmatique simulée ».

Pour avancer davantage vers l'étude concrète de la compréhension des références, nous avons cherché à réaliser un système opérationnel, qui permette l'étude expérimentale de phénomènes liés à la référence dans les textes. Nous avons pour cela conçu un modèle représentationnel, en liaison avec le projet CERVICAL ; ce modèle peut être appliqué à des agents linguistiques supposés capables de maintenir, dans une mémoire, des représentations structurées des entités, par exemple sous la forme d'ensembles $\{<attribut_i, valeur_i>\}$.

Nous avons vu au Chapitre V que le décalage entre les propositions théoriques et les systèmes réalisés restait élevé : pratiquement tous les systèmes évalués pour la résolution de la référence dans les textes étaient fondés sur des heuristiques nettement moins élaborées que les modèles computationnels. Dans ces conditions, nous avons pris le parti de concevoir un système qui soit *robuste* et *évaluable*, donc fondé sur une version simplifiée de notre modèle représentationnel. Un système plus élaboré aurait été handicapé par le manque de ressources, disponibles à une échelle insuffisante.

Nous décrirons d'abord le modèle proposé pour la référence (VII.B.1), ainsi que sa particularisation au dialogue homme-machine et à la compréhension des textes narratifs (VII.B.2). Dans les deux cas, le système manipule des « représentations mentales » des référents : la section VII.B.3 en décrit le principe et introduit une terminologie compatible avec les considérations théoriques du V.D. Nous comparons brièvement cette approche avec d'autres systèmes (VII.B.4.iii), avant de détailler l'algorithme que nous proposons pour la résolution de la référence (VII.C.3), ainsi que les structures de données (VII.C.1) et les opérations correspondantes (VII.C.2). La

distance entre notre modèle théorique et son implémentation pourra paraître notable, mais les résultats du système montreront l'intérêt pratique du modèle lui-même.

Nous passons ensuite à une partie plus technique, mais indispensable à l'exécution et à l'évaluation du programme : la constitution de ressources, c'est-à-dire de textes suffisamment longs, pour lesquels « la bonne solution » a été établie, afin de pouvoir procéder à l'évaluation (VII.D). Les méthodes d'évaluation sont rapidement résumées au VII.E, puisque, à cause de la complexité de la question, nous lui consacrons un développement substantiel (troisième partie de notre thèse). Nous présentons aussi l'Atelier de Traitement de la Référence (VII.F) qui intègre les outils réalisés ; en particulier, nous décrirons un optimiseur pour les paramètres numériques du résolveur (VII.F.2).

Enfin, de nombreux résultats numériques sont donnés : meilleurs scores (VII.G.1), comparaison de plusieurs variantes de l'algorithme (VII.G.2), résultats sur la mémoire à court terme du système (VII.G.3), et sur l'optimisation (VII.G.4). Enfin, nous montrons en détail comment l'évaluation permet une mesure de la pertinence individuelle de chaque règle donnée au système (VII.G.5).

VII.B Cadre théorique

VII.B.1 Aperçu du modèle

Nous concevons ici la situation de communication d'un point de vue représentationniste[1]. Un locuteur, appelé aussi émetteur e, et un interlocuteur ou récepteur r partagent un même monde ou univers U, vu comme un ensemble d'entités ayant diverses propriétés. Ces entités peuvent être matérielles ou conceptuelles, et peuvent appartenir à des constructions fictives[2].

Chaque individu possède sa propre perception du monde extérieur, notée ici $p_e(U)$ et respectivement $p_r(U)$; on a bien sûr $p_e(U) \neq p_r(U)$. La représentation que chaque individu possède du monde est le résultat de ses perceptions p et des inférences $i \circ p$ qu'il peut faire sur celles-ci, utilisant ses connaissances et croyances antérieures. On note ces représentations RU_e et RU_r ; on note alors :

$RU_e = i_e(p_e(U)) = ip_e(U)$ et aussi $RU_r = i_r(p_r(W)) = ip_r(U)$.

[1] Cette partie théorique a fait l'objet d'une présentation de notre part à la conférence Coling-ACL'98 (Popescu-Belis, Robba et Sabah 1998).

[2] La distinction d'un émetteur et d'un récepteur peut paraître arbitraire dans une situation de communication à double sens. S'il est vrai que l'on pourrait aussi se contenter d'appeler les individus locuteur-A et locuteur-B, nous préférons néanmoins la distinction émetteur/récepteur, qui s'avérera pertinente dans le cas de la compréhension de récits, lorsque la communication est à sens unique. Par ailleurs, nous avons commenté la notion *d'entité*, par rapport à celle de *propriété* ou *action*, au VI.B.3.i.

Pour des raisons de vraisemblance, il est utile de considérer que seule une partie du monde a un rôle dans cette situation de communication ; nous l'appelons *sujet* ou *domaine* de la situation, et les locuteurs en possèdent les représentations RS_e et RS_r. Ce cadre est représenté sur la Figure VII-1 ci-dessous.

L'émetteur produit un message discursif *MD*, et un message gestuel (ou déictique) *MG*. Les deux messages contiennent des *expressions référentielles* (en abrégé, dans toute la suite, ER). Les ER sont des fragments de discours ou des gestes qui sont liés à des représentations d'entités situées dans les RS_e et/ou RS_r. Ainsi, chaque *RS* (ou bien *RU*) contient un ensemble de *représentations mentales* des entités du monde et de leurs propriétés (en abrégé, dans toute la suite, RM).

Chaque individu construit également des représentations des *RU* des autres individus, utilisant bien sûr toujours ses perceptions et inférences. Ainsi, l'émetteur possède sa propre représentation RU_e, mais aussi une $RU_{e(r)} = ip_e(RU_e)$; le récepteur possède RU_r mais aussi $RU_{r(e)} = ip_r(RU_e)$. On peut bien sûr continuer indéfiniment une telle hiérarchie, avec $RU_{r(e(r))}$, $RU_{r(e(r(e)))}$, etc. Par analogie avec la théorie des jeux, on peut appeler cette hiérarchie *spécularité*, et dire que lorsque toutes les *RU* de tous les interlocuteurs deviennent identiques pour un sujet donné (donc les *RS*), alors ce sujet devient un « connaissance mutuelle ».

Un message discursif ou gestuel (*MD* ou *MG*) a été compris si, pour le sujet de la situation de communication, $RS_{r(e)} = RS_e$, à savoir si la représentation que se fait le récepteur de la représentation du monde propre à l'émetteur est exacte. Il s'agit bien sûr d'une simplification, dans une perspective informationnelle, pour pouvoir proposer un modèle computationnel. Dans une perspective plus rhétorique, on peut dire par exemple qu'une communication réussit si RS_r change selon les intentions de l'émetteur *e*. Dans la même visée simplificatrice, nous ne représentons pas l'évolution dans le temps, donc nous n'indexons pas les différentes représentations par un indice temporel.

Afin de comprendre un message (*MD* ou *MG*), le récepteur doit associer les expressions référentielles du message (linguistiques ou gestuelles) aux entités existantes ou construits dans sa représentation du monde. Pour ce faire, le récepteur utilise sa propre représentation du monde RU_r, ou plus exactement celle du sujet RS_r, afin de construire d'abord, et mettre à jour ensuite, ses représentations des entités (RM) à partir des expressions référentielles (ER).

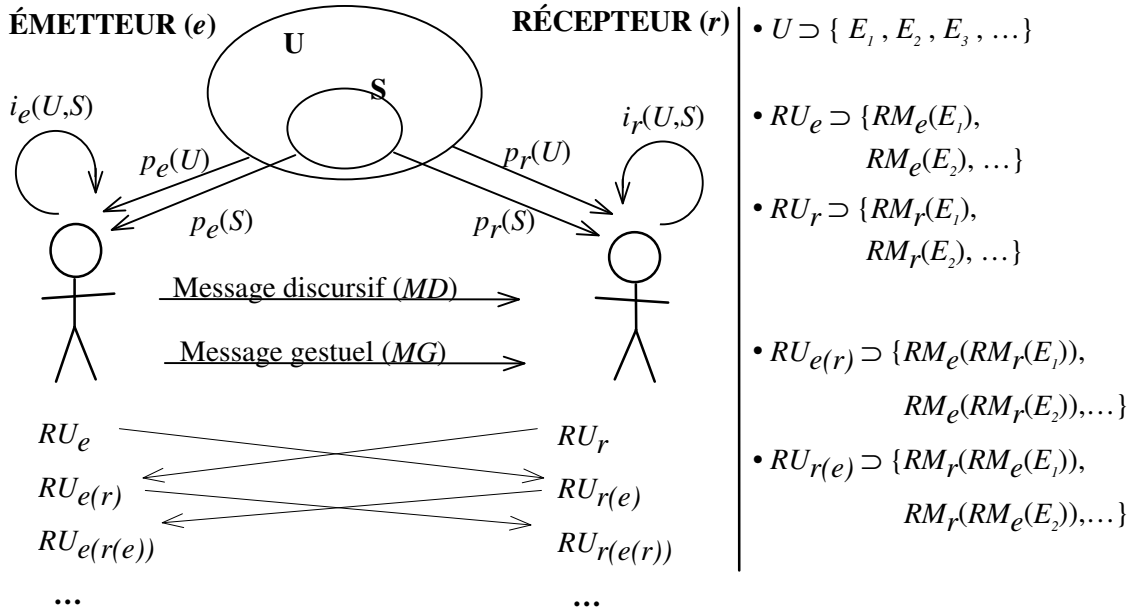

Figure VII-1. Le schéma proposé pour la représentation de la référence.

VII.B.2 Application du modèle à deux cas particuliers

En ce qui concerne le problème qui nous intéresse ici, à savoir la compréhension des références par une méthode computationnelle ou un programme informatique (noté *c*), le formalisme précédent se particularise aisément. En toute généralité, un tel programme doit construire et gérer sa propre représentation du monde RU_c (ou du sujet de la communication RS_c), ainsi que celle de l'utilisateur ($RU_{c(s)}$ ou $RS_{c(s)}$), à l'aide de ses connaissances sur le monde (sa représentation du monde) et du message lui-même (texte, pointages avec la souris, capture d'images, etc.).

VII.B.2.i Le dialogue homme-machine

Les applications de commande en langage naturel sont un premier cas particulier important – par exemple un gestionnaire de fenêtres (la thèse de B. Gaiffe (1992) étudie la référence pour ces systèmes). Le sujet *S* est constitué par les objets graphiques affichables à l'écran, le reste de l'univers *U* n'intervenant pas en principe. La perception que le programme a du sujet est alors totalement exacte, puisque le programme contrôle l'écran : $p_c(S) = S$; par conséquent, $p_c(S)$ est la source d'information la plus importante et la plus fiable. Les pointages avec la souris fournissent directement l'information déictique. La différence entre RS_c et $RS_{c(s)}$, i.e. entre la représentation exacte de la situation et celle que semble avoir l'utilisateur (telle qu'elle ressort de ses messages), doit être attribuée à la différence entre la description complète des objets graphiques et leurs traits visibles à l'écran.

VII.B.2.ii La compréhension de textes narratifs par un programme

Les données sont toutes autres pour un programme chargé de la compréhension de textes écrits, et plus particulièrement des récits – il s'agit du cas que nous développerons par la suite. Pour un tel programme, comme d'ailleurs pour l'humain lisant un récit, il ne peut y avoir de perception directe du sujet décrit S : à la fois $p_c(S)$ et le message gestuel MG sont absents. Le lecteur humain s'appuie dans ce cas, pour construire sa RS_r, seulement sur l'information donnée par le récit, et aussi, naturellement, sur sa connaissance du langage et du monde ; ces connaissances doivent évidemment être partagées avec le narrateur pour que la compréhension soit possible.

Le programme suivra le même principe ; s'appuyant sur le récit, il construira d'abord une représentation qui dépend entièrement de celle du narrateur : $RS_{c(e)}$. Si le programme est suffisamment élaboré, il peut alors à l'aide d'un raisonnement supplémentaire construire sa propre représentation du monde RS_c. La différence entre $RS_{c(e)}$ et RS_c peut être due à un emploi incohérent des ER par le locuteur, ou bien à des contradictions internes de l'histoire, voire à une intention du narrateur de tromper le lecteur, détectée par le programme.

Il va sans dire que de telles subtilités semblent hors de portée des programmes actuels, si les récits présentés ont une formulation libre. Notre système de suivi des références dans les récits, décrit ci-dessous, se limitera à la construction d'une représentation « immédiate » $RS_{c(e)}$ des référents du texte. Pourtant, le formalisme que nous proposons permet de représenter le côté référentiel de situations relativement complexes ; nous donnons en annexe (XI.B) un exemple d'application à la modélisation d'une scène de théâtre à cinq personnages. Cette représentation peut être reliée aux travaux sur la modélisation des « points de vue » dans les textes narratifs (Berthelin 1979).

VII.B.3 Le paradigme des représentations mentales pour la résolution de la référence

Nous nous tournons maintenant plus particulièrement vers le problème de la référence. Le but de la résolution de la référence est à nos yeux la construction de représentations mentales (RM) des entités ou référents de l'univers du discours, par les participants à la situation de communication. Les représentations mentales (sous-entendu celles des référents) doivent être construites et actualisées en fonction des perceptions et des messages reçus, et en particulier de leurs expressions référentielles (ER). Dans le cas de la compréhension de textes narratifs, les RM doivent être construites en utilisant le texte lui-même ; les connaissances sur l'environnement commun sont importantes, alors que les capacités perceptives semblent inutiles.

L'état de l'art du Chapitre V a montré que la prise en compte des représentations que les interlocuteurs se font des entités du discours était indispensable à une description correcte du phénomène de la référence. Nous avons vu comment l'idée de « fiche » introduite par Karttunen (V.B.2.iv et (Karttunen 1976)) avait été reprise ou suggérée

indépendamment dans de nombreux travaux postérieurs, comme ceux d'Evans ou Récanati, pour les plus avancés – mais qui se trouvent très loin d'une quelconque implémentation. En revanche, notre but ici n'est pas tant de proposer un cadre théorique nouveau, mais de réaliser un *dénominateur commun implémentable* des travaux antérieurs, et montrer que l'adoption, pour la réalisation informatique, d'un cadre ontologiquement vraisemblable permet aussi d'obtenir des résultats concrets significatifs.

Faisons à ce stade un point terminologique, synthétisé sur la Figure VII-2 ci-dessous. La relation entre chaque RM_i et l'entité associée E_k est appelée *représentation* ; on dit aussi que la RM_i *évoque* l'entité E_k, pour choisir un terme plus neutre. Une expression référentielle ER_j *spécifie* une certaine RM_i, et parvient ainsi à *(se) référer à* l'entité E_k qui est évoquée par cette RM_i. La relation entre deux ER qui évoquent la même RM, donc qui se réfèrent à la même entité, est appelée *co-spécification* ou simplement *coréférence*. Ces termes sont analogues à ceux que propose Sidner (cf. V.D.2.i.a).

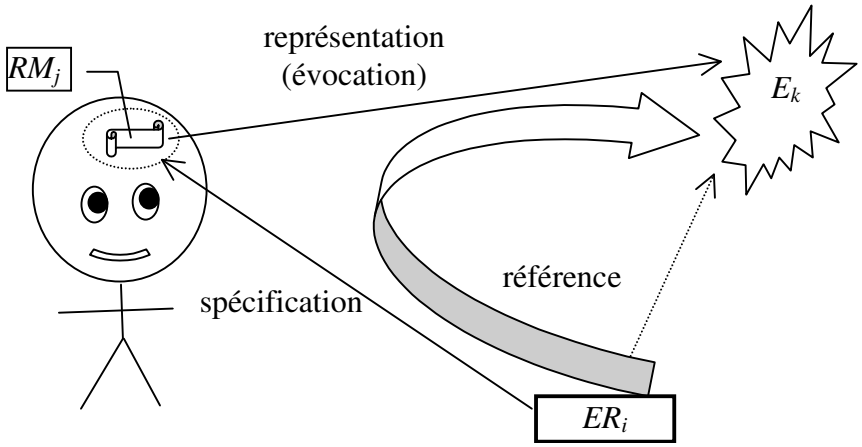

Figure VII-2. Représentation des termes utilisés dans notre modèle.

Nous avons adapté ces considérations théoriques aux capacités techniques disponibles, réalisant ainsi un système avec un ensemble réduit de connaissances, mais qui reste conforme au « paradigme des représentations mentales » – et qui est aussi une version personnelle, adaptée et simplifiée des conclusions plus abstraites et plus ouvertes du projet CERVICAL (cf. V.D.5). Notre implémentation détaillée plus bas maintient deux caractéristiques essentielles du « paradigme des RM » : concrètement, nos RM sont des ensembles d'ER, et les traits caractéristiques de chaque RM sont obtenus en faisant une moyenne des traits de toutes les ER composantes. En outre, chaque RM possède une activation propre dans l'ensemble total, ce qui modélise la saillance du référent à l'esprit d'un récepteur humain, et caractérise le *groupe* d'ER correspondant.

VII.B.4 Comparaison avec d'autres approches

VII.B.4.i Les RM et la résolution de la *co*référence

Le « paradigme des RM », pas plus que l'implémentation que nous en proposons, ne peuvent se réduire à la simple résolution de la coréférence entre paires d'ER. En effet, même si la résolution de la coréférence permet de retrouver les ensembles d'ER correspondant aux RM (en parcourant les liens coréférentiels), cette construction se fait seulement *a posteriori* et passe à côté des deux propriétés dynamiques spécifiques à nos RM : *l'activation* et *la moyenne des traits*. Ces propriétés du modèle sont centrales même dans notre implémentation « plus faible » du paradigme. Nous allons montrer quelques avantages importants qu'il y a à regrouper les ER dans des RM plutôt que de construire des liens de coréférence entre paires d'ER

- Premièrement, la résolution de la coréférence comporte une ambiguïté inhérente et superflue : si ER_1 et ER_2 sont déjà connues comme coréférentes (notons cela $ER_1 \sim ER_2$), alors il n'y a pas de différence conceptuelle, pour une troisième ER, entre $ER_3 \sim ER_1$ et $ER_3 \sim ER_2$, et donc ces deux possibilités ne devraient pas être traités comme *une* alternative[3]. En d'autres mots, il faut tirer profit du fait que la relation de coréférence est une relation d'équivalence, par conséquent transitive. La question de savoir si ER_i et ER_j sont coréférentes demande un calcul complexe si l'on utilise les liens de coréférence, alors qu'elle très simple si l'on utilise des RM ou ensembles, car ceux-ci permettent de stocker de façon commode toute l'information déjà acquise.

- Deuxièmement, si l'on recherche seulement des liens potentiels de coréférence pour l'ER courante, quelles ER antérieures doit-on examiner ? Visiblement, il faudrait examiner seulement les groupes nominaux pleins, car les pronoms ne réfèrent en principe qu'à travers des GN, et sont insuffisamment contraints (« saturés ») pour servir seuls d'antécédents potentiels. Nous avons toutefois montré que cette dichotomie n'était pas défendable (cf. V.C.2 et les travaux d'Ariel (1990) et de Reboul (1994)), et qu'il y avait un continuum entre ces deux pôles. Le niveau conceptuel approprié n'est donc pas celui des ER isolées, mais celui des RM, c'est-à-dire des ensembles d'ER déjà résolues.

- Troisièmement, les liens de coréférence ne peuvent pas représenter des dépendances multiples, de types variés, qui sont nécessaires pour représenter des relations référentielles plus complexes, comme par exemple tout/partie (abordées au VII.D.2.iv ci-dessous). Les liens de coréférence signalent simplement l'identité du référent (co-

[3] Même si, à un niveau de représentation plus fin, la nature ou la « connotation » des ER peut rendre ces choix non équivalents.

spécification) pour les deux ER, alors que la gestion des collections d'entités, elles-mêmes des entités, nécessite des liens typés.

Comme nous l'avons montré au chapitre précédent, les systèmes de traitement de la référence utilisent dans leur grande majorité les liens de coréférence, comme par exemple la plupart des systèmes présents aux évaluations MUC-6 (1995), ou encore le système de Kennedy et Boguraev (1996). Dans ce dernier cas, les liens de coréférence sont une généralisation des liens entre antécédent et anaphorique – utilisés pour la résolution des pronoms – vers lesquels nous nous tournons maintenant.

VII.B.4.ii Les RM et l'opposition anaphorique/antécédent

L'approche défendue ici diffère aussi de l'opposition antécédent/anaphorique. Celle-ci postule des liens de coréférence asymétriques entres les anaphoriques, en général des pronoms, et leurs antécédents, en général des groupes nominaux « pleins », ou « sémantiquement saturés ». Cette séparation des ER en deux classes est artificielle, nous venons de le dire ; lorsque l'on va des pronoms vers les descriptions définies et les noms propres, de plus en plus de connaissances puisées dans le discours lui-même sont nécessaires pour résoudre les ER. Néanmoins, le principe fondamental de la tâche de résolution – le traitement de chaque ER – est le même pour toutes les ER.

Nous avons montré dans un travail plus ancien[4] que l'utilisation du « paradigme des RM » apportait une amélioration notable à la résolution des anaphores elles-mêmes. Nous ne donnons pas ici le détail de ce travail, pour plusieurs raisons. Le mécanisme de résolution était une version préliminaire, moins fondée théoriquement, du mécanisme présenté ci-après. D'autre part, nos travaux sur l'évaluation de la résolution de l'anaphore mettent en question la pertinence des mesures utilisées dans ce premier travail : la notion de « nombre de pronoms *correctement* rattachés à leur antécédent », pourtant très répandue et paraissant de bon sens, est contestable du moment où l'antécédent n'est pas unique (cf. le Chapitre IX et surtout le IX.E.4 pour une discussion détaillée).

L'idée du travail évoqué était de comparer les succès de notre système de résolution de l'anaphore dans deux cas : avec ou sans résolution des coréférences entre autres GN pleins (et leur regroupement). D'après notre système de comptage, le regroupement des GN pleins coréférents permettait de passer de 40% à 60% environ de pronoms correctement résolus ; le travail montrait donc le bien fondé du « paradigme RM » appliqué à la résolution des anaphores. Encore une fois, dans la mesure où ces chiffres nous paraissent maintenant attaquables, nous ne les reprendrons pas ici.

[4] Ce travail réalisé en collaboration avec Isabelle Robba a été présenté à un atelier de la conférence ACL'97/EACL'97 (Popescu-Belis et Robba 1997).

VII.B.4.iii Fondements théoriques de quelques autres systèmes

VII.B.4.iii.a Systèmes munis de connaissances de haut niveau

Le système LaSIE (Gaizauskas et Humphreys 1997, Gaizauskas, Wakao, Humphreys, Cunningham et Wilks 1995), présenté aux évaluations MUC-6 utilise une représentation des référents fondée sur des connaissances sémantiques de haut niveau. Par conséquent, le bon fonctionnement du système est très dépendant de la tâche – alors que nous cherchons un système plus robuste. D'autre part, l'idée d'activation des RM que nous utilisons a semblé inutile aux créateurs de LaSIE (Azzam, Humphreys et Gaizauskas 1998) ; il est vrai que si l'on dispose des informations sémantiques suffisantes, un facteur pragmatique comme l'activation peut être inutile dans le cas de textes simples. Autre exemple de système utilisant une représentation des référents, celui de Susan Luperfoy (1992) manipule des « crochets discursifs » (« discourse pegs »), c'est-à-dire des pointeurs auxquels on rattache les différentes descriptions des objets. Ces représentations ont été utilisées dans un système de dialogue homme-machine, mais aucune évaluation numérique ne semble disponible – alors que nous souhaitons réaliser un système évaluable. À un niveau plus théorique, le modèle d'Appelt et Kronfeld (1987) (cf. V.A.4.ii.b) est par certains aspects proche du nôtre (sauf en ce qui concerne l'utilisation d'actes de langage), mais son adaptation pour l'implémentation ne semble pas aisée, surtout pour la compréhension de textes narratifs.

VII.B.4.iii.b Systèmes robustes, connaissances de bas niveau

La plupart des systèmes robustes adoptent le paradigme de la *co*référence, tel qu'il est défini aussi par la tâche de coréférence MUC-6 (1995) et MUC-7 (1998). De nombreuses techniques robustes, « de bas niveau », ont été développées, utilisant en général des techniques d'appariement (« pattern matching ») entre les chaînes de caractères potentiellement coréférentes, par exemple (Fisher, Soderland, McCarthy, Feng et Lehnert 1995). Des techniques d'apprentissage automatique ont même été utilisées pour induire les meilleures règles d'appariement (McCarthy et Lehnert 1995). Une technique intéressante et originale a été proposée par Dekang Lin (1995), utilisant la satisfaction des contraintes pour grouper les ER en *ensembles* d'ER coréférentes ; cette idée s'accorde avec le paradigme des RM, mais ne peut pas fonctionner de façon incrémentale en même temps que les facteurs d'activation.

VII.C Principes d'implémentation

Nous donnons maintenant un modèle d'implémentation du cadre théorique exposé, en nous fondant sur les outils linguistiques disponibles déjà, ou qui le seront à moyen terme, dans notre environnement de recherche. L'implémentation réalisée ne peut satisfaire à toutes les spécifications, certains des outils linguistiques n'étant pas encore opérationnels ou suffisamment robustes. Nous rendrons compte dans l'exposé, chaque fois que cela sera nécessaire, de ce que nous avons implémenté (voir aussi la section VII.F sur l'atelier de traitement de la référence, et les interfaces en annexe XI.C).

VII.C.1 Structures de données

Nous donnons d'abord les spécifications de notre modèle pour les structures associées aux ER et aux RM, sans tenir compte des limites dues à l'implémentation. La structure de données représentant une ER devrait posséder les champs suivants :

- *ER.Identificateur* – index de l'ER dans l'ensemble total ;
- *ER.Position* – identification exacte de la position de l'ER : paragraphe, phrase, position des mots initial et final ;
- *ER.InformationSyntaxique* – une représentation syntaxique attachée à l'ER, et sa fonction. Mieux, si elle est disponible, la représentation syntaxique de la phrase entière contenant l'ER (un pointeur sur celle-ci) ;
- *ER.InformationSémantique* – une structure conceptuelle pour l'ER, ou, si elle est disponible, pour la phrase entière ; cela suppose l'existence aussi d'un dictionnaire conceptuel indépendant ; en fonction des connaissances disponibles, ce dernier attribut peut manquer totalement.

Notre système ne dispose pas encore de représentations sémantiques, puisque l'analyseur sémantique développé dans notre groupe, fondé sur les graphes conceptuels, n'est pas encore suffisamment robuste (Vapillon, Briffault, Sabah et Chibout 1997). Toujours pour des raisons de robustesse, nous ne pouvons utiliser avec confiance les arbres syntaxiques fournis par l'analyseur LFG[5] que pour les seules ER, et non des phrases complètes. Il s'agit plus exactement de structures de traits ou *f-structures* (LFG).

Les attributs de la structure de données attachée aux RM dépendent aussi des capacités de traitement. En voici une liste, version simplifiée de celle que propose le projet CERVICAL (V.D.5) :

- *RM.Identificateur* – numéro d'index unique ;
- *RM.Liste(ER)* – la liste des ER qui se réfèrent, dans le texte traité, à l'entité évoquée par la RM ;
- *RM.InformationSémantiqueDepuisLeTexte* – une structure conceptuelle qui accumule les propriétés sémantiques particulières ou conjoncturelles de l'entité, en utilisant les ER résolues et les phrases où celles-ci apparaissent (champ *ER.Information-Sémantique*) ;
- *RM.InformationSémantiqueDepuisLeDictionnaire* – une structure conceptuelle qui accumule les propriétés sémantiques de l'entité, telles qu'elles ressortent du dictionnaire conceptuel du système ; ces propriétés reflètent les connaissances *a*

[5] LFG : Lexical Functional Grammar, ou Grammaire Lexicale Fonctionnelle, voir par exemple (Abeillé 1993:ch.1) pour une description.

priori (ou encyclopédiques) sur la catégorie sémantique (ou classe) à laquelle appartient l'entité ;

- *RM.Relations* – les relations *typées* avec d'autres RM, comme par exemple *partie_de/composé_de*, *type_de/instance_de*, *fonction_de/individu_exerçant* ; le premier type, peut-être le plus important, permet la gestion des RM plurielles ;

- *RM.PointeurVersObjetInfo* – un pointeur vers l'entité évoquée, si celle-ci appartient à une autre application accessible au système (cas du dialogue de commande), par exemple une fenêtre ou menu ;

- *RM.InformationPerceptive* – extension des informations sémantiques, si le système peut gérer les représentations perceptives, surtout visuelles, ce qui permet d'utiliser des considérations contextuelles et déictiques pour la résolution.

Nous préconisons, pour la représentation de l'information sémantique, l'usage des graphes conceptuels de John Sowa (1984, 1992). Ces représentations utilisent une hiérarchie (treillis) de concepts et un ensemble de graphes canoniques associés aux concepts pour représenter le « sens » d'un énoncé sous la forme d'instances de concepts, reliées entre elles par des instances de relations. Un analyseur conceptuel utilisant ces graphes est en cours de développement dans notre groupe (Vapillon, Briffault, Sabah et Chibout 1997).

Pour les RM que nous décrivons ici, l'information sémantique (graphe conceptuel) provenant du texte est mise à jour grâce à l'appariement (suivi d'une éventuelle adjonction) de chaque ER que le programme attache à la RM. L'information sémantique encyclopédique est stockée sous forme de pointeurs vers le treillis des concepts et les graphes canoniques du concept et de ses sur-concepts. Le mécanisme d'appariement ou unification des graphes sert aussi de base aux contraintes de compatibilité référentielle entre ER et RM, détaillées au VII.C.3.iv.

Nous avons conçu et réalisé une implémentation nécessitant peu de connaissances, pour pallier les problèmes de robustesse des outils linguistiques. En particulier, la structure sémantique d'une RM n'est pas construite, et nous utilisons à sa place l'ensemble des analyses syntaxiques des ER composant la RM, grâce à un mécanisme original décrit au VII.C.3.iii. Parmi les attributs précédents, *seuls les deux premiers sont donc utilisés*. Nous avons étudié l'attribut *RM.Relations* pour la représentation du pluriel dans l'annotation de textes (cf. VII.D.2.iv ci-dessous) et pour l'évaluation (cf. IX.E.5). Pour l'heure, l'information concernant la RM est donc stockée dans la liste d'ER qui la composent : on peut même dire que la RM n'est autre chose que cet ensemble.

VII.C.2 Opérations

Nous donnons ici quelques opérations de base sur l'ensemble des RM : les trois premières sont classiques et indispensables, alors que les deux dernières reprennent une idée du projet CERVICAL en vue du traitement du pluriel.

- création : $ER_i \rightarrow RM_{nouvelle}$; une nouvelle RM est créée lorsqu'une ER spécifie pour la première fois une RM (ou nouvelle entité) ;
- attachement : $ER_i + RM_a \rightarrow RM_a$; lorsqu'une ER spécifie une RM déjà existante, l'ER est attachée à la RM et les données de la RM sont mises à jour ;
- fusion : $RM_a + RM_b \rightarrow RM_{nouvelle}$; il peut apparaître à un moment donné que deux RM ont été construites pour ce qui est en réalité la même entité, on doit donc les fusionner. L'opération symétrique, à savoir la séparation d'un RM qui représente en réalité deux entités, est beaucoup plus difficile, car de très nombreuses décisions doivent être révisées, donc nous ne la considérerons pas pour l'instant ;
- partition : $RM_a \rightarrow RM_a + RM_{nouvelle(1)} + RM_{nouvelle(2)} + \ldots$;
- groupement : $RM_a + RM_b \rightarrow RM_a + RM_b + RM_{nouvelle(a,b)}$;

Les deux dernières opérations, partition et groupement sont duales et font intervenir des entités composées d'autres entités. Par exemple, à partir d'un nom collectif comme *l'équipe de football* le programme devrait utiliser ses connaissances sur les éléments d'une équipe pour créer plusieurs RM correspondant aux joueurs – et résoudre ainsi correctement l'ER *le premier joueur*. Réciproquement, après avoir par exemple construit deux RM pour *Mademoiselle X* et *Madame Y*, le programme devrait construire aussi la RM groupement, et être capable ainsi à résoudre l'ER *les deux femmes*. Dans les deux cas, l'attribut *RM.Relations* doit être convenablement rempli avec des pointeurs (typés) vers les RM conteneur, respectivement contenu.

L'opportunité de réaliser une partition ou un groupement a fait l'objet de travaux dans le projet CERVICAL (Reboul *et al.* 1997). Dans l'exemple de *l'équipe de football*, on peut imaginer une partition qui crée onze nouvelles RM (si on ignore les remplaçants), mais une telle création ne devrait pas être automatique, comme on le voit avec :

La 27ème Division Alpine part en manœuvres. Le brigadier X nous raconte : …

Inversement, le groupement peut aussi aboutir à une multiplication inutile des RM (irréaliste et pénalisante pour la résolution), donc ne doit pas être automatique[6].

[6] Un stage d'ingénieur effectué dans sous notre direction a mis en évidence les difficultés du sujet (Jan 1998).

VII.C.3 Algorithme de résolution

VII.C.3.i Identification vs. traitement des ER

Dans son ensemble, nous divisons en deux phases le processus de résolution de la référence dans un texte (donné à l'avance). Tout d'abord, on doit localiser et reconnaître les expressions référentielles ER ; puis, on doit à partir de celles-ci et du reste du texte construire les représentations mentales RM spécifiées par le texte.

Notre pensons que c'est la deuxième étape qui constitue à proprement parler l'essence de la résolution de la référence. En effet, même si la première n'est pas dépourvue de difficultés, elle peut néanmoins être menée à bien informatiquement de façon assez élémentaire, en sélectionnant par exemple tous les groupes nominaux. Il est alors probable que l'on aura sélectionné aussi des noms qui ne réfèrent pas (par exemple *prendre congé*, etc.). Ce sera donc le rôle de la deuxième étape de leur associer ou non une RM. En d'autres termes, puisqu'il est quasiment impossible de trouver les termes qui réfèrent réellement sans résoudre complètement la référence, on doit accepter comme première phase une sélection moins stricte des ER, en vue de la construction finale de RM, le cas échéant.

Cela appelle deux remarques supplémentaires. La première, c'est que le caractère référentiel semble pouvoir être considéré comme graduel, ainsi que l'ont montré les arguments cités au VI.B.3. La seconde, c'est que la construction d'une RM pour une ER qui en réalité n'en spécifie aucune n'est pas pénalisée par la plupart des mesures d'évaluation (Chapitre IX), pourvu bien sûr qu'on n'attache pas d'autres ER à cette RM.

VII.C.3.ii Description d'ensemble du traitement des ER

Les ER du texte à traiter sont examinées une par une. Pour chacune d'entre elles, notre résolveur doit choisir entre la création d'une nouvelle RM, ou le rattachement de l'ER à une RM existante, qu'elle spécifierait. La méthode choisie ici, qui s'inspire de plusieurs algorithmes récents (vus au chapitre précédent) procède en deux étapes. D'abord, le programme détermine l'ensemble des RM qui pourraient être spécifiées par l'ER traitée, en appliquant un certain nombre de contraintes de compatibilité (nous en verrons au VII.C.3.iii les particularités, liées à notre représentation des RM).

Puis, en utilisant le degré d'activation que possède chaque RM (VII.C.3.v), le programme choisit parmi les RM sélectionnées la plus active. C'est elle que l'ER courante est supposée spécifier, et elle lui est rattachée. Si aucune des RM compatibles n'est suffisamment active (le seuil est fixé expérimentalement), et *a fortiori* s'il n'y a aucune RM compatible, une nouvelle RM est créée à partir de l'ER courante. À la fin de ce *cycle*, l'activation des RM est recalculée en fonction de l'opération réalisée ; puis, l'ER suivante est examinée. Un tel cycle est représenté sur la Figure VII-3 ci-dessous[7].

[7] Les détails de l'implémentation (réalisée en langage Smalltalk) sont donnés dans l'annexe XI.C page 301.

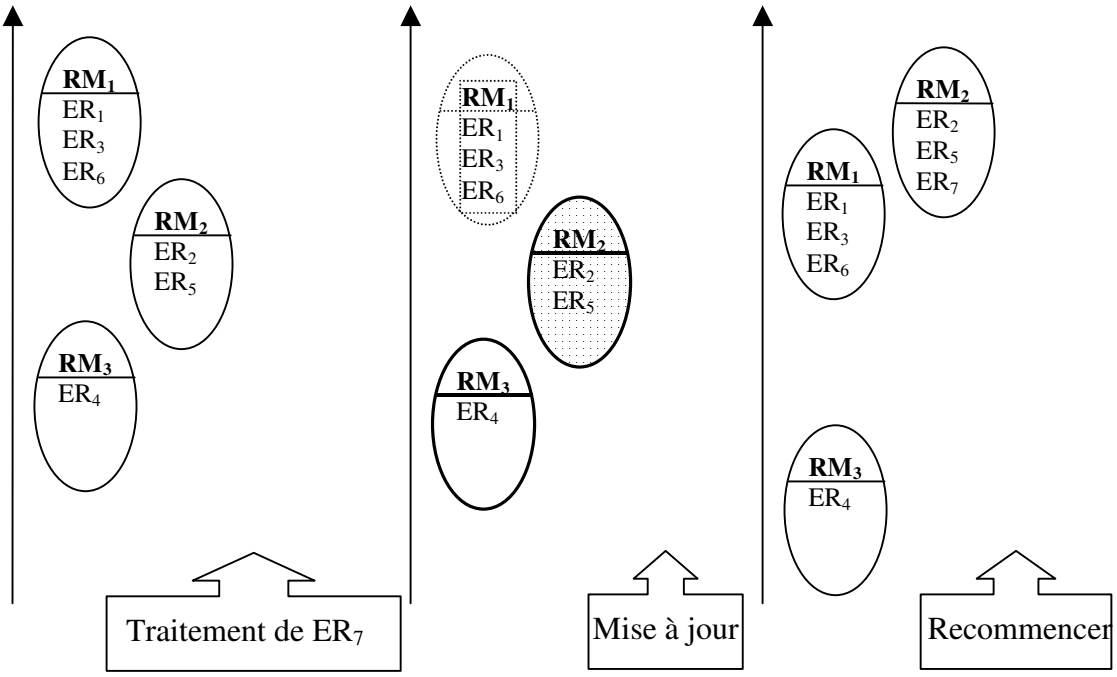

Figure VII-3. Représentation d'un cycle de fonctionnement du résolveur de références : lecture d'une nouvelle ER → contraintes de compatibilité → rattachement à la RM la plus active si elle existe (au centre, RM_2) → mise à jour des activations.

VII.C.3.iii Heuristiques de compatibilité

Lors de la première étape de chaque cycle de résolution, le programme doit sélectionner le sous-ensemble des RM qui sont susceptibles d'être spécifiées par l'ER courante, sur des bases syntaxiques et sémantiques. Dans la mesure où une RM est caractérisée par sa liste d'ER (*RM.liste(ER)*), la possibilité que l'ER courante spécifie une RM se traduit par des contraintes exprimées entre l'ER courante et la liste des ER de cette RM. On calcule cela à partir de contraintes plus élémentaires de compatibilité référentielle entre couples d'ER (VII.C.3.iv).

L'intégration des contraintes entre une ER et un ensemble d'ER fait l'objet de différentes *heuristiques de compatibilité* ; il s'agit à notre connaissance d'un point original, qui provient de notre volonté de conserver le paradigme des RM tout en n'utilisant que des connaissances linguistiques élémentaires. Si les outils permettaient de construire pour chaque RM une structure sémantique fiable, alors la compatibilité entre une ER et une RM s'exprimerait par des contraintes d'appariement sur leurs graphes conceptuels respectifs (*RM.InformationSémantiqueDepuisLeTexte* et *ER.InformationSémantique)*.

Nous avons examiné différentes heuristiques. Une des plus élémentaires est :

(H1) [La RM_a est susceptible d'être spécifiée par l'ER_i] *ssi*

[ER_1 étant la 1ère ER de $RM_a.Liste(ER)$, ER_i et ER_1 peuvent co-spécifier][8]

Cette heuristique est fondée sur l'idée que la première ER qui spécifie une RM possède quelque chose de typique (souvent vrai, mais pas toujours).

Afin d'utiliser pleinement le paradigme des RM, il peut paraître plus judicieux d'examiner la compatibilité de l'ER courante avec chacune des ER constituant la RM. En réalité, cela n'est pas pertinent pour *toutes* les ER constituant la RM. Le cas le plus manifeste est celui des ER pronominales, qui n'ont pas d'influence sur la compatibilité sémantique[9]. Nous allons donc exclure les ER pronominales de cette heuristique de compatibilité, dans notre implémentation robuste, tout en rappelant que la distinction entre pronoms et GN pleins est en théorie critiquable (V.C.2 et (Ariel 1990, Reboul 1994)). Une deuxième heuristique de compatibilité est donc :

(H2) [La RM_a est susceptible d'être spécifiée par l'ER_i] *ssi*

[$\forall ER_j \in RM_a.Liste(ER)$ non pronominale, ER_i et ER_j peuvent co-spécifier]

Cette heuristique se révèle inefficace car elle est trop contraignante. D'abord, elle ne permet qu'une faible variation dans l'appellation d'une entité. Surtout, elle néglige la distinction déjà évoquée (**V.D.4**) entre la fonction d'identification d'une ER (spécifier une RM) et sa fonction informative (ajouter de l'information à une RM). Si deux ER intègrent des informations différentes sur la même RM, et donc semblent ne pas être coréférentes, l'heuristique précédente risque d'invalider le rattachement de la seconde à la RM de la première (par exemple) même s'il y a compatibilité référentielle avec de nombreuses autres ER de la RM.

L'heuristique « opposée » du point de vue de la tolérance est donc :

(H3) [La RM_a est susceptible d'être spécifiée par l'ER_i] *ssi*

[$\exists ER_j \in RM_a.Liste(ER)$ non pronominale, ER_i et ER_j peuvent co-spécifier]

Enfin, nous pouvons énoncer une heuristique moyenne des deux heuristiques précédentes H2 (« toutes ») et H3 (« au moins une ») :

(H4) [La RM_a est susceptible d'être spécifiée par l'ER_i] *ssi*

[Pour plus de X% des $ER_j \in RM_a.Liste(ER)$, ER_i et ER_j peuvent co-spécifier]

On peut choisir ici d'imposer où non que ces ER soient non pronominales. Lorsque X varie de 0 à 100%, cette heuristique varie de H3 à H2, et les états intermédiaires peuvent être testés. H3 peut sembler plus proche du paradigme des liens de coréférence, puisqu'elle signifie le rattachement de l'ER courante à la première ER compatible de la RM. Cette analogie est toutefois inexacte ici, puisque les RM sont aussi caractérisées par un coefficient d'activation qui évolue avec le discours, coefficient qui

[8] Les contraintes pour que deux ER puissent co-spécifier font donc l'objet du VII.C.3.iv. Nous notons *ssi* le connecteur *si et seulement si*.

[9] Une ER pronominale véhicule à strictement parler l'information sémantique associée au prédicat dont elle est un argument ; cette information se retrouve en théorie dans le champ *RM.InformationSémantique-DepuisLeTexte*, mais ici nous ne disposons pas d'une telle structure.

ne peut pas être représenté dans le paradigme des liens. Aucune des heuristiques ci-dessus ne diminue donc la spécificité de notre implémentation.

VII.C.3.iv Compatibilité référentielle entre deux ER

Les heuristiques précédentes mettent en œuvre des critères ou contraintes de compatibilité entre deux ER ; dans la mesure où notre système doit s'appliquer à des textes non restreints, seules des règles très élémentaires sont utilisées. Il y a en tout trois règles :

- accord en genre entre les deux ER (règle morpho-syntaxique) ;
- accord en nombre entre les deux ER (règle morpho-syntaxique) ;
- accord « sémantique » entre les ER. La compatibilité des têtes et des modifieurs des GN est vérifiée à l'aide d'un réseau sémantique pour le moment très limité. Nous utilisons en effet un dictionnaire de synonymes/hyperonymes/hyponymes adapté aux principaux concepts rencontrés dans notre corpus, mais cela serait utilement remplacé par une ressource plus étendue, un réseau de type WordNet (Fellbaum 1998) pour la langue française, comme par exemple le dictionnaire de Ploux et Victorri (1998).

VII.C.3.v Activation des RM

La valeur d'activation de chaque RM reflète sa proéminence ou saillance dans le texte, à chaque moment. Le concept d'activation a été évoqué dans certains travaux linguistiques sur la référence (V.C.2), mais son application au traitement computationnel de la référence est le mieux représentée par les travaux d'Alshawi (1987), et le système de Lappin et Leass (1994) (V.D.2.ii et V.E.2). Ces travaux proposent un certain nombre de facteurs d'activation.

Les facteurs que nous utilisons ici pour mettre à jour l'activation des RM sont :

- lors de sa *création*, chaque RM reçoit une activation initiale ;
- lors du *rattachement* d'une ER à une RM, l'activation de la RM augmente (*réactivation*) en fonction de la nature de l'ER rattachée. Les cas distingués sont : GN ou nom propre, GN ou nom commun, pronom ;
- le passage du temps (l'avancement dans le texte) entraîne pour chaque RM une baisse de l'activation, selon le nombre de mots, phrases, paragraphes et unités thématiques écoulés[10].

[10] Voici un exemple effectivement utilisé : ActivationInitiale=15 ; ActivationParNomPropre=40, ActivationParAutreSNouN=20, ActivationParPronom=10 ; BaisseActivationParUT=8, BaisseActivation-ParParagraphe=4, BaisseActivationParPhrase=2, BaisseActivationParMot=1 ; TailleMémoireRM=15, TailleListeERcaract=10 (pour ce dernier paramètre, voir VII.G.3).

Le résolveur examine l'activation des RM référentiellement compatibles avec l'ER courante (VII.C.3.iv ci-dessus) ; si l'activation maximale est faible, une nouvelle RM est créée à partir de l'ER courante, sinon l'ER est rattachée à la RM compatible la plus active. L'activation est donc un facteur dynamique qui dépend de l'ensemble des décisions prises par le programme dans son interprétation du texte.

Nous avons ajouté un paramètre auxiliaire relié à l'activation, à savoir une taille maximale pour le nombre de RM gardées en mémoire. Lorsque l'activation d'une RM devient nulle, elle sort de cette mémoire à court terme ; mais si trop de RM ont une activation non nulle, cette mémoire ne garde que les plus actives (cf. aussi les résultats au VII.G.3 ci-dessous. Cette « mémoire » est l'analogue de la mémoire de travail d'un interlocuteur humain, dont la capacité à gérer simultanément de nombreux référents est limitée. D'un point de vue computationnel, cette limitation accélère la sélection des RM compatibles avec l'ER courante.

Pour plus de réalisme, il faudrait aussi modéliser une archive plus permanente de RM (une « mémoire à long terme ») dont les éléments pourraient repasser en mémoire de travail lors d'une réactivation suffisamment forte. À ce niveau pourrait être étudiée l'intégration dans l'architecture CARAMEL du LIMSI (Sabah 1990, Sabah et Briffault 1993), prenant en compte le modèle de la mémoire que propose (Bassi Acuña 1995).

VII.D Construction de ressources réutilisables pour la résolution de la référence

La réalisation effective d'un résolveur de références dans les textes dépend de l'existence de ressources linguistiques permettant la mise au point, la vérification, l'entraînement et l'évaluation du système[11]. Pour les deux premières tâches, des textes courts peuvent être facilement élaborés par le programmeur, mais les deux dernières nécessitent des textes longs pour lesquels la solution correcte a déjà été définie.

De telles ressources sont encore très rares. Les campagnes d'évaluation MUC, ayant proposé une tâche de résolution des coréférences, ont utilisé des ensembles d'articles en *anglais*, traitant d'un thème fixé pour chaque campagne. Ces nombreux textes ont été balisés par les organisateurs avec les coréférences correctes, en vue de la comparaison avec les réponses des participants ; mais ces textes ne sont pas publiquement disponibles. Pour le français, le serveur Silfide (LORIA, Nancy) fournit, en accès restreint, un texte d'environ 30 000 mots dont les « personnages principaux » sont balisés – il s'agit du début du *Père Goriot* de Balzac (cf. VII.D.4), désormais noté LPG..

VII.D.1 L'alternative RM ou *co*référence pour les ressources

La construction de ressources cohérentes et facilement réutilisables demande une définition explicite des conventions d'annotation, outre la définition du phénomène

[11] Les considérations exposées dans cette section ont fait l'objet d'une publication (Popescu-Belis 1998).

annoté. L'opposition RM/coréférence introduite plus haut (VII.B.4.i) se retrouve à propos des conventions d'annotation.

Les conférences MUC adoptent l'approche « coréférence ». La description de la tâche de coréférence pour MUC-6 (1995) ainsi que sa révision pour MUC-7 (Hirschman 1997), contiennent des indications détaillées pour les ER et les liens de coréférence à annoter. Nous avons montré ci-dessus pourquoi la notion de « groupement d'ER coréférentes » était plus pertinente d'un point de vue computationnel que celle de « liens de coréférence ». Pour l'annotation, les deux possibilités sont à première vue logiquement équivalentes, puisque on peut passer indifféremment des liens aux classes et *vice versa*. Il y a néanmoins trois différences importantes :

- 1° Le cas des RM formées d'une unique ER : les instructions MUC ne préconisent pas le balisage de ces ER/RM, car elles n'interviennent pas dans des liens de coréférence, bien qu'elles réfèrent tout autant que les autres à des entités (le protocole d'évaluation MUC n'en tient pas compte, cf. IX.C.1 et IX.E.2). Toutefois, Hirschman *et al.* (1998) ont suggéré de scinder l'annotation en deux phases : d'abord le marquage de toutes les ER, puis le marquage des liens de coréférence.

- 2° Le balisage des liens de coréférence est plus sujet aux erreurs que le balisage des RM, dans la mesure où l'annotateur doit choisir une ER précise de rattachement parmi de nombreuses ER « antécédentes », alors que le nombre réel de RM est bien plus réduit.

- 3° Les chaînes de coréférence ne peuvent pas représenter facilement différents types de relations référentielles. Certes, la tâche MUC-7 (Hirschman 1997) se concentre sur la coréférence de type « identité », à savoir les ER qui désignent strictement la même entité. Mais les relations de type tout/partie, classe/instance, ont aussi leur intérêt, même si leur fréquence est faible, et devraient être annotées dans la perspective d'une utilisation future de la ressource. Nous verrons ci-dessous comment l'annotation des RM permet de gérer ces relations.

VII.D.2 Trois niveaux d'annotation et leurs étiquettes

Nous avons défini pour nos ressources trois niveaux d'annotation, chacun ajoutant au texte des balises supplémentaires par rapport au précédent. Ces niveaux sont conçus essentiellement pour le balisage de textes narratifs, en vue de la constitution de « réponses correctes » ou clés, auxquelles on compare les « réponses » d'un système de résolution de la référence. Selon les objectifs de l'évaluation, on pourra exiger d'un système de partir du texte brut, et réaliser automatiquement toutes les annotations, ou bien lui fournir un niveau intermédiaire ; la réponse est par définition au dernier niveau.

VII.D.2.i Niveau I : structure élémentaire du discours

Le premier niveau consiste à marquer des indications de structuration essentielles pour la résolution de la référence. Un texte est délimité par <tt> et </tt>, et se divise en sections ou « unités thématiques », délimitées par <ut> et </ut> (Ferret 1998, Ferret et Grau 1998, Ferret, Grau et Masson 1998). Ces unités représentent des zones de « cohérence pragmatique » élevée, qui peuvent être déterminées par des méthodes de segmentation automatique. Les unités thématiques se composent de paragraphes (,) et de phrases (<s>, </s>).

Ces marquages suffisent pour des textes narratifs, qui contiennent peu d'objets complexes (par exemple des tables) ou des titres. Un point qui reste délicat est celui des dialogues dans le style direct, qui se rencontrent par exemple dans le texte LPG (Silfide). Pour ce texte, le balisage manuel effectué marque le locuteur de chaque réplique à l'aide de noms standard inclus dans les balises ; par exemple, une balise ouvrante sera <u id="25" who="Rastignac">. Ces informations sont essentielles pour une résolution correcte des pronoms de première et deuxième personne dans les dialogues, et plus généralement pour de nombreux pronoms déictiques. Elles sont néanmoins très difficiles à produire automatiquement, c'est pourquoi nous n'en tiendrons pas compte dans la suite. Pour ce faire, il y a deux possibilités : ne résoudre que des textes sans dialogues au style direct, ou ne résoudre que les pronoms de troisième personne, et considérer les première et deuxième personnes comme des déictiques ou indexicaux.

VII.D.2.ii Niveau II : marquage des ER

Plusieurs balises ont été proposées pour la délimitation des expressions référentielles. Les campagnes MUC utilisent une balise <coref> qui permet de gérer aussi les coréférences, alors que Bruneseaux et Romary (1997) proposent <rs> comme « referring string » (chaîne de caractères qui réfère). Bien sûr, le nom exact importe peu, car il est facile d'en réaliser la conversion : nous utilisons <er>. Notre balise comporte un seul attribut, qui est le numéro d'index, unique, de chaque ER, par exemple <er id="25">la jeune femme</er>. Le point essentiel est qu'il faut baliser à ce stade *toutes* les ER du texte, et non seulement celles qui sont coréférentes avec d'autres.

Les balises MUC, conçues pour l'évaluation compétitive, permettent une certaine variation dans l'étendue d'une ER. Ainsi, la « bonne réponse » marque pour chaque ER les étendues maximale et minimale qui doivent être marquées dans la réponse pour que l'ER soit considérée comme repérée. Concrètement, cela se traduit par un marquage avec les balises <coref>, </coref> de la chaîne maximale constituant l'ER, et l'introduction d'un champ « min » dans <coref> qui a pour valeur une copie de la chaîne minimale caractérisant l'ER.

Nous ne préconisons pas l'usage d'un tel attribut, qui est spécifique à la définition d'une clé pour une compétition (les systèmes ne sont pas supposés le marquer). En échange, il faut naturellement définir l'extension exacte des ER balisées : nous suggérons de prendre en compte *l'ensemble du groupe nominal avec ses modifieurs, mais sans les phrases relatives*. De toute façon, pour évaluer la résolution de la référence en soi, il faut

fournir au système des textes *déjà annotés* au niveau II, et lui demander de regrouper les ER en RM ; dans ce cas, clé et réponse possèdent les mêmes ER.

VII.D.2.iii Niveau III : marquage des RM

Les RM ne figurent pas directement dans le texte, donc ne peuvent faire l'objet d'un marquage « sur le texte » : elles seront représentées comme des relations entre ER.

Les conventions MUC expriment des liens de coréférence entre ER utilisant un attribut « ref » dans la balise ouvrante <coref>. La valeur de cet attribut est le numéro d'index d'une ER qui coréfère avec l'ER balisée. Il est à remarquer qu'un seul attribut « ref » est autorisé, et ce afin d'obtenir des chaîne de coréférence, et non des graphes. Ce marquage est probablement le plus économique, car il n'introduit pas de nouvelle balise. En voici un exemple, où l'on n'a marqué que les ER intervenant dans des relations de coréférence (nous ne traitons pas encore les possessifs, ni les réflexifs) :

<er id=1>*Félix*</er> *prit seulement son épée.* <er id=2 ref=1>*Il*</er> *était sur le point de sortir quand* <er id=3 ref=2>*il*</er> *trouva sa mère Camille,* <er id=4>*toutes les femmes de la maison*</er>*, et parmi* <er id=5 ref=4> *elles*</er> *Vittoria.*

Les conventions adoptées par le LORIA (Bruneseaux 1998, Bruneseaux et Romary 1997) préconisent la construction de structures regroupant les ER à l'aide de nouvelles balises. Cela conduit à une gestion plus transparente des liens, et ouvre la voie vers l'annotation de relations référentielles plus complexes, entre RM. Voici le même texte que ci-dessus :

<er id=1>*Félix*</er> *prit seulement son épée.* <er id=2>*Il*</er> *était sur le point de sortir quand* <er id=3>*il*</er> *trouva sa mère Camille,* <er id=4>*toutes les femmes de la maison*</er>*, et parmi* <er id=5>*elles*</er> *Vittoria.*

```
<linkGrp type="coref">
<link target="1 2">
<link target="2 3">
</linkGrp>
```

```
<linkGrp type="coref">
<link target="4 5">
</linkGrp>
```

Ces structures apparaissent à la fin du texte, ou à la fin d'une section si le texte est long. Dans ce dernier cas, le lien entre deux structures représentant la même RM mais séparées par une ou plusieurs sections se fait en rattachant la première ER de la seconde à la dernière ER de la première, le lien figurant dans la seconde RM. Ce mécanisme nuit quelque peu à l'homogénéité de l'annotation, et est sujet à erreurs, mais permet de séparer les références aux personnages principaux en groupes de taille raisonnable, .

Nous n'introduisons pas d'autre annotation pour les RM, car ces deux conventions suffisent à ce stade. Notre système permet de lire et d'écrire des ressources aux deux formats, et utilise pour la résolution un format interne objet. Ces annotations ne sont jamais manipulées à la main, mais à travers des interfaces de saisie.

VII.D.2.iv Différents types de relations

Les chaînes de coréférence MUC ne permettent pas de représenter aisément différents types de relation référentielles : tout/partie, classe/instance, variable/valeur, individu/fonction – puisqu'une ER ne peut établir qu'un seul lien (attribut "ref"). Les conventions LORIA, malgré l'attribut "type" dans la balise "linkGrp", ne semblent pas être appliquées dans le texte LPG. En effet, dans cette ressource, les relations de type tout/partie qui sous-tendent les RM plurielles sont représentées par une suite d'indices dans la balise "link", par exemple :

```
<link target="6 5 9"></link>
```

Cela signifie que la RM dont fait partie l'ER n°6 est composée des RM dont font partie les ER n°5 et n°9 (cf. texte correspondant ci-dessous). Le premier indice possède donc un statut particulier. Cette convention, on le voit, n'est pas aisée à manipuler et ne permet pas d'exprimer le type de la relation référentielle. Les annotateurs eux-mêmes précisent d'ailleurs qu'il ne s'agit que d'un choix temporaire – mais la ressource LPG a été publiée telle quelle, et cette convention a été utilisée environ 30 fois sur 100 pages de texte.

Nous proposons la solution suivante, qui a l'avantage de construire une nouvelle structure pour chaque RM plurielle, et de la même façon pour chaque RM de type « classe » ou « catégorie » (la structure suit le paragraphe).

```
<er id=1>Félix</er> prit seulement <er id=2>son épée </er>. <er id=3>Il</er>
était sur le point de sortir <er id=4>il</er> trouva <er id=5>sa mère
Camille</er>, <er id=6>toutes les femmes de <er id=7>la maison</er></er>,
and parmi <er id=8>elles</er> <er id=9>Vittoria</er>.
  <linkGrp type="part_of">
  <link target="6 5">
  <link target="6 9">
  </linkGrp>
```

Il s'agit donc de typer correctement les différents liens figurant dans les balises "link" ; ici, le typage "part_of" rend ces liens asymétriques, par rapport au typage "coref".

VII.D.3 Nos outils et traitements pour l'annotation de la référence

Nous donnons maintenant les méthodes que nous avons effectivement employées pour annoter nos textes, méthodes que nous estimons adaptées à l'annotation d'autres textes. Nous divisons l'annotation en deux phases, d'abord le balisage des ER, puis leur regroupement en classes ou RM ; cette division est aussi préconisée par (Hirschman, Robinson, Burger et Vilain 1998), avec pour but d'augmenter l'accord entre annotateurs. Elle correspond aussi aux étapes de la résolution de la référence sur des textes non balisés.

VII.D.3.i Segmentation

Nous réalisons à présent la segmentation en paragraphes et phrases en utilisant des indices élémentaires comme les retours-chariot ou la ponctuation, suffisants pour nos textes narratifs. Cela peut être amélioré dans le cas général, la segmentation étant un problème assez complexe pour des textes contenant des objets complexes (par exemple séparés par des points...). Nous avons effectué des essais pour réaliser automatiquement la segmentation en unités thématiques – système disponible dans notre groupe (Ferret 1998, Ferret et Grau 1998, Ferret, Grau et Masson 1998) – mais la précision obtenue laisse encore à désirer. En attendant le perfectionnement du système, les volumes de textes à baliser rendent tout à fait possible le balisage manuel.

VII.D.3.ii Annotation des ER

Dans une expérience préliminaire, nous avons testé le repérage automatique des ER à l'aide de l'analyseur LFG développé dans notre groupe (Vapillon, Briffault, Sabah et Chibout 1997). Néanmoins, sa trop grande ambiguïté dans la délimitation des groupes nominaux (nos ER) nous a conduit d'abord à réaliser une interface de choix et validation manuels des ER proposées[12]. Cette interface (cf. Figure XI-3 en annexe) permet de sélectionner une partie des syntagmes proposés par l'analyseur comme satisfaisant des critères donnés à l'avance (ex. : GN, pronom, etc.). La lenteur du processus nous a toutefois conduit à l'abandonner au profit d'un balisage manuel ; un analyseur plus robuste pourrait le rendre au moins semi-automatique.

Nous avons donc choisi d'insérer les balises <er>, </er> manuellement dans le texte, à l'aide d'un simple éditeur. Cette conception minimaliste illustre en réalité la simplicité de la tâche, qui s'exécute à l'aide de simples clics souris. La vérification de la bonne syntaxe des balises SGML (ouverture/fermeture et imbrication) est faite automatiquement par un analyseur que nous avons réalisé ; ce même programme introduit des indices dans les balises afin d'indexer les ER, ce qui servira à leur regroupement en RM.

Le problème à ce stade est la définition de critères homogènes et cohérents de délimitation des ER. Comme annoncé, nous avons choisi de baliser l'intégralité de chaque GN, mais sans les éventuelles propositions relatives. Une question délicate est celle des GN non référentiels ; si on désire baliser les ER « strictes », il ne faut pas baliser les nominaux non référentiels. Le problème est souvent délicat à cause des expressions idiomatiques, et ne semble soluble par une machine dans l'immédiat. C'est pourquoi nous avons choisi de baliser tous les GN, i.e. toutes les expressions « d'apparence référentielle », quitte à ne les rattacher, au niveau III, à aucune autre ER (elles donnent lieu à des RM singleton).

[12] Cette ambiguïté est due au fait que, dans l'état actuel, les modules de niveau supérieur censés désambiguïser les résultats de l'analyseur (Sabah 1996, Sabah et Briffault 1993) ne sont pas encore opérationnels.

VII.D.3.iii Analyse syntaxique

Une fois les limites des ER balisées, nous les transmettons à l'analyseur syntaxique LFG afin de leur ajouter une *f-structure* (structure de traits LFG). Pour pallier l'ambiguïté de l'analyseur, nous avons développé une autre interface qui permet de choisir manuellement la *f-structure* correcte parmi celles proposées (ou du moins celle qui s'en rapproche le mieux, car parfois le GN n'est pas analysé en entier) ; des données numériques sur ces problèmes figurent dans la Table VII-2 ci-dessous. Naturellement, la plupart de ces informations sont difficiles à représenter sur le texte à l'aide de balises autour des ER ; nous les sauvegardons dans un format objet spécifique à notre système, avec l'ensemble du texte, des ER et des RM correctes.

VII.D.3.iv Construction des RM

Une interface pour le regroupement aisé des ER en RM a été conçue par Isabelle Robba (Popescu-Belis et Robba 1998). L'opération est réalisée uniquement par des clics souris à l'aide desquels on rattache l'ER courante à une RM nouvelle ou existante. L'interface ne gère pas encore les relations typées (tout/partie, classe/instance). L'interface permet de traiter environ 700 ER par heure.

Nous avons écrit pour notre part un générateur de balises SGML qui produit des textes balisés SGML à partir de l'objet informatique produit par l'interface précédente. Ainsi, le balisage est syntaxiquement correct par construction – tout un ensemble d'erreurs provenant du balisage manuel sur le texte étant évitées. Par exemple, sur le texte LPG, nous avons observé (et rapporté aux annotateurs) soit des liens manquants (la RM apparaît alors comme deux RM distinctes ou plus), soit des liens excédentaires, dus à une faute de frappe dans les indices des ER (deux RM sont alors fusionnées). Bien sûr, la génération des balises SGML n'exclut pas un rattachement manuel erroné d'une ER à une RM.

VII.D.3.v Vérification

Outre l'exemple précédent, le protocole de constitution de ressources que nous avons défini permet d'autres vérifications aussi. En effet, l'idée de travailler sur des objets Smalltalk qui sont convertis depuis et vers le format SGML par des modules automatiques permet d'éviter des erreurs de détail (dans le balisage), et permet de vérifier les ressources importées.

Ainsi, après la vérification syntaxique, les nombres d'ER dans le texte et dans les RM (la clé) sont comparés, afin de vérifier que toutes les ER ont été réparties en RM, et qu'il n'y pas de duplication des ER dans les RM. De même, les index des ER et RM sont vérifiés, et au besoin rétablis afin d'obtenir une suite naturelle. Ces vérifications concernent surtout les ressources importées.

La vérification manuelle de la correction des RM est très importante. Hirschman *et al.* (1998) ont développé une interface qui montre les ER coréférentes sur une même ligne d'un tableau, les colonnes correspondant aux phrases. Cette interface ne convient pas à nos textes, car ils sont beaucoup trop longs pour que le tableau soit lisible (certaines

RM étant présentes du début jusqu'à la fin du texte). Pour vérifier de façon conviviale la cohérence de nos RM, nous avons modifié le générateur SGML pour produire du texte HTML sur lequel les ER correspondant à une RM donnée (personnage) apparaissent en couleur. Il suffit alors d'ouvrir le texte HTML dans un navigateur et de le parcourir en suivant les ER colorées ; pour simplifier, on ne peut colorer qu'une ER à la fois. Cette méthode nous semble la plus adaptée, étant donnée que notre texte le plus long (LPG) possède environ 3800 ER, et les personnages principaux sont évoqués par des centaines d'ER, réparties sur toute la longueur du texte.

VII.D.4 Textes utilisés

Nous avons effectué au début de notre travail une étude comparative sur trois textes, au vocabulaire relativement peu spécialisé, afin de déterminer le domaine le plus riche en phénomènes référentiels. Pour ce faire, nous nous sommes intéressé aux pronoms, puisque ceux-ci se prêtent plus facilement à la détection automatique. Les textes sont :

- un texte narratif, extrait des *Chroniques Italiennes* de Stendhal, intitulé *Vittoria Accoramboni* (en abrégé VA) ;
- un texte de rapport scientifique écrit par Gérard Sabah pour le rapport scientifique annuel du LIMSI ;
- une suite d'articles boursiers extraits du journal *Le Monde*.

Les résultats numériques pour les diverses catégories de pronoms étudiées sont donnés dans la Table VII-1 ci-dessous. Ces résultats montrent clairement que la densité de pronoms est de loin la plus élevée pour le texte de Stendhal (VA), donc désigne celui-ci comme le plus intéressant à étudier. C'est en effet dans ce texte qu'on retrouve le plus clairement des « personnages » (donc des RM) qui ont une bonne continuité au long du texte.

Comme nous l'avons dit, peu de ressources balisées pour la coréférence sont accessibles en français. La seule dont nous avons connaissance et que nous utilisons est le texte LPG (serveur Silfide, LORIA, Nancy), à savoir premier chapitre du roman *Le Père Goriot* de Balzac. Sur ce texte sont annotées les ER et RM correspondant seulement aux personnages principaux, humains, lieux, objets (sauf le début où tout est annoté) – conventions de (Bruneseaux et Romary 1997).

	Stendhal	Sabah	Le Monde
• nombre de mots	9 144	15 006	16 504
• *il* pronom personnel *il* impersonnel	165 44	50 45	35 45
• *elle* pronom personnel	31	19	25
• *le* article défini *le* pronom clitique	217 16	226 7	367 3
• *la* article défini *la* pronom clitique	201 7	345 2	577 1
• *l'* article *l'* pronom clitique masculin *l'* pronom clitique féminin	95 13 7	374 6 4	242 2 0
• *lui* objet indirect masculin *lui* objet indirect féminin *lui* pronom tonique masculin	37 10 20	5 1 2	3 5 6
• *son*, *sa*, *ses* possessifs	110	46	87

Table VII-1. Données quantitatives sur la présence des pronoms dans trois textes de natures différentes (narration, rapport scientifique, articles boursiers).

La vérification du texte LPG par nos outils nous a permis de corriger quelques erreurs dues au travail manuel ; nous avons aussi ajouté des *f-structures* aux ER. Nous avons isolé dans LPG une partie initiale notée LPG.eq, de taille équivalente au texte de Stendhal (VA). Nous avons balisé toutes les entités de ce dernier au LIMSI (avec Isabelle Robba), et ajouté des *f-structures*.

	VA	*LPG.eq*	*LPG*
Nombre de mots	2630	7405	28576
Nombre d'ER	638	686	3359
Nombre de RM (clé)	372	216	480
Rapport ER / RM	1.72	3.18	7.00
ER nominales	510	390	1864
ER pronominales	102	262	1398
ER non analysées	26	34	97

Table VII-2. Caractéristiques référentielles des trois textes balisés.

Des données numériques sur ces trois textes sont données dans la Table VII-2. On observe que le taux de coréférence (rapport ER/RM) augmente de VA à LPG, en passant par LPG.eq. Le taux de coréférence important de ces deux derniers est dû à la limitation du balisage aux personnages principaux, auxquels on réfère souvent. On observe aussi

sur ces données les problèmes liés à l'analyseur syntaxique : le silence vaut environ 4% , et plus grave, l'ambiguïté est en moyenne de 2,7 *f-structures* par ER, ce qui justifie notre interface de choix manuel.

Malgré ces difficultés, nous avons continué à travailler sur des textes narratifs longs et complexes afin d'évaluer en détail, et de manière quantitative, les effets des différentes règles et paramètres de résolution de la référence là où la densité (co)référentielle est très grande. Il est vraisemblable que la résolution atteint de meilleurs scores sur des textes plus simples, comme la documentation technique ou les articles de journal, puisque les entités y reçoivent des noms plus figés.

VII.E Aperçu des méthodes d'évaluation

VII.E.1 Introduction

L'évaluation des performances d'un système de résolution de la référence sur des textes narratifs peut sembler simple à première vue. Une fois que des humains sont tombés d'accord sur le balisage correct du texte (la clé), il faut demander au programme de baliser les RM qu'il a trouvées (la réponse) puis on compare ces deux annotations.

Cette comparaison doit être mesurable, et obéir à des critères précis, reproductibles. Vu la longueur des textes, elle doit pouvoir être implémentée dans un programme d'évaluation automatique. Un tel programme mesurerait donc sans peine la proximité de la clé et de la réponse, ce qui permettrait : (1) l'évaluation fréquente et aisée d'un système ; (2) la réalisation d'optimisations automatiques (VII.F.2 et VII.G.4 ci-dessous) ; (3) l'estimation de la contribution de chaque règle de compatibilité entre ER à la performance générale du système (VII.G.5).

La comparaison des ensembles de RM clé et réponse par un algorithme n'est en réalité pas triviale : elle constitue en soi un problème théorique sérieux qui a rarement été abordé. Nous étudions en détail cette question dans la troisième partie de notre thèse (Chapitre IX), c'est pourquoi nous nous contenterons ici d'une brève description des différentes mesures proposées.

VII.E.2 Concepts essentiels

La comparaison de la clé et de la réponse s'exprime différemment selon que l'on adopte la perspective « par liens » ou « par classes ». La première a été utilisée aux conférences MUC et a le mérite de poser le problème de façon intuitive. Si on admet que le but du système de résolution de la référence est de trouver les liens corrects entre les différentes ER, on peut décomposer la réponse en trois parties (comme en recherche d'information (Salton et McGill 1983, Van Rijsbergen 1979)) : (1) les liens corrects trouvés par le système ; (2) les liens erronés supposés par le système ; (3) les liens corrects non trouvés par le système. On définit la *précision* comme la proportion de (1) dans (1)+(2), le *rappel* comme la proportion de (1) dans (1)+(3), et la *f-mesure* comme la

moyenne harmonique du rappel et de la précision. On normalise en général ces valeurs pour obtenir un succès de rappel et de précision compris entre 0 et 100%.

Nous verrons ultérieurement pourquoi cette description intuitive, d'apparence claire, est en fait très problématique (cf. IX.B.2). Pour l'heure, on peut remarquer que ce ne sont pas les liens exacts qui sont pertinents, mais les RM (ensembles d'ER connectées) ; en effet, plusieurs configurations de liens peuvent conduire aux mêmes RM. Et de fait, les concepteurs de la méthode d'évaluation MUC ont cherché à construire un algorithme indépendant de la répartition exacte des liens. Cet algorithme s'avère trop indulgent, et plusieurs adaptations en ont été proposées.

Dans la perspective « par classes », on remarque d'abord que les RM constituent une partition de l'ensemble des ER : elles sont disjointes et recouvrent cet ensemble, et ce sont aussi des classes d'équivalence pour la relation de coréférence entre ER. Le problème de l'évaluation revient alors à estimer les différences entre deux partitions d'un même ensemble. Il n'existe cependant pas, en mathématiques, une mesure quantitative simple applicable à cette question. C'est pourquoi nous avons étudié différentes possibilités, résumées ci-dessous ; l'étude extensive est donnée au Chapitre IX.

VII.E.3 Résumé des mesures d'évaluation

VII.E.3.i Mesures existantes

VII.E.3.i.a La mesure MUC

Le rappel et la précision MUC (notés *MRS* et *MPS*) correspondent à une interprétation maximalement indulgente des définitions données ci-dessous. Le rappel MUC compte comme erreurs de rappel le nombre minimal de liens que l'on estime manquants pour reconstituer les classes clé à partir de la réponse (et inversement pour la précision). Cette indulgence est d'autant plus visible que le taux de coréférence des textes augmente.

VII.E.3.i.b La mesure B^3

La mesure proposée par Bagga et Baldwin (1998a, 1998b) tente de répondre à l'indulgence de la mesure MUC en pénalisant les fusions de classes volumineuses, qui peuvent être le fruit d'une tactique triviale. Les scores obtenus sont en général (mais pas toujours) plus faibles que ceux de MUC.

VII.E.3.i.c Extension de MUC à l'aide du « κ »

R. Passonneau (1997) propose de calculer, à partir des scores MUC, les probabilités que deux annotateurs tombent d'accord ou non sur la présence, respectivement l'absence, d'un lien de coréférence. On mesure ainsi la proximité des deux annotations en tenant compte des possibilités d'accord par hasard, à l'aide du coefficient κ (Krippendorff 1980), compris entre 1 (accord total) et -1 (opposition totale), 0 représentant l'indépendance. On peut appliquer aussi ces résultats à la comparaison de

la clé et de la réponse. Les résultats sont moins indulgents que MUC, mais moins informatifs (un chiffre au lieu de deux).

VII.E.3.ii Nos propositions

Nous avons proposé plusieurs réponses aux limitations précédentes, restant plus proches du paradigme des RM, i.e. des classes d'ER. Ces propositions sont développées au IX.D et IX.G.

VII.E.3.ii.a La mesure C (noyaux)

Pour cette mesure, on identifie pour chaque RM clé la RM réponse qui lui correspond le mieux (qui contient le maximum d'ER de cette RM clé), qui sera appelée le noyau de la RM clé. On compte alors comme erreur de rappel toute ER en dehors du noyau (son inverse, le succès, est noté *CRS*). On procède de façon symétrique pour la précision (notée *CPS*). Cette mesure est toujours plus sévère que la mesure MUC.

VII.E.3.ii.b La mesure XC (noyaux exclusifs)

L'idée de cette mesure est d'interdire à deux classes clé d'avoir le même noyau, ce qui demande un algorithme assez complexe pour construire des « noyaux exclusifs ». On compte, pour chaque RM clé, les ER hors du noyau exclusif comme erreur de rappel, et les ER erronées dans ce noyau comme erreur de précision (on ne construit pas de noyaux exclusifs symétriques pour les classes réponse). Cette mesure est souvent plus sévère que la précédente, mais assez peu parlante.

VII.E.3.ii.c Les coefficients distributionnels

Nous avons défini aussi deux coefficients qui mesurent la proximité du profil distributionnel des ER en RM, pour la clé et la réponse. Ce profil est obtenu en prenant les tailles des RM par ordre décroissant. Le *recouvrement* mesure la similarité algébrique entre les deux profils, et varie de 0 à 100% – une valeur de 100% signifie que les RM ont mêmes tailles, ce qui n'assure pas qu'elles sont identiques. Réciproquement, une valeur différente de 100% signale que la clé et la réponse diffèrent. La *d-erreur* permet de savoir si les ER sont plus regroupées dans la clé que dans la réponse (*d-erreur* de rappel) ou inversement (*d-erreur* de précision) ; sa formule est assez complexe. L'intérêt de cette mesure est d'indiquer la direction prioritaire dans laquelle on doit améliorer son système : améliorer la précision ou le rappel.

VII.E.3.ii.d La mesure H (entropique)

Nous avons élaboré une dernière mesure fondée sur un modèle théorique reconnu, la théorie de l'information. Nous y reviendrons en détail au IX.G, et nous n'utiliserons pas ses résultats numériques dans ce chapitre. Bien que solidement fondée et bénéficiant de propriétés théoriques convenables, cette mesure (valeurs numériques données au IX.E) n'apporte pas de changement qualitatif par rapport aux précédentes.

VII.E.4 Conclusion sur les mesures

Pour conclure brièvement sur ce point, nous nous bornerons à souligner qu'aucune mesure ne semble s'imposer aux dépens des autres ; par conséquent, il est utile de conserver les résultats de plusieurs mesures, et d'avoir une confiance accrue dans ces résultats lorsque les mesures varient dans le même sens. C'est-à-dire, une amélioration de l'ensemble des mesures traduit, avec une bonne certitude, l'amélioration d'un système donné, sur un texte donné.

VII.F L'Atelier de Traitement de la Référence

VII.F.1 Vue d'ensemble

L'ensemble des programmes décrits jusqu'ici – résolveur, modules de conversion, interfaces de saisie – s'intègre dans une architecture (Figure VII-4) permettant la communication entre modules, dans l'environnement de programmation objet du Groupe Langage et Cognition du LIMSI (langage Smalltalk, environnement Envy).

La partie centrale de l'architecture est le gestionnaire des « unités de traitement » (interface en annexe XI.C, Figure XI-1). Ces unités sont des structures de données (des objets Smalltalk) qui regroupent une représentation structurée d'un texte, la liste des ER correspondantes avec leurs *f-structures*, la liste des RM correctes (clé), et éventuellement la liste des RM produites par le résolveur (réponse). Ces unités de traitement sont les structures qui permettent à la fois la résolution et son évaluation, lorsqu'elles sont complètes.

Notre système peut traiter des ressources aux niveaux I, II ou III, et aussi « 0 », à savoir des textes non balisés. Nous avons décrit dans la section VII.D.3 ci-dessus la transition d'un niveau à l'autre, qui nécessite divers outils ou interfaces d'annotation manuelle. Aux différentes extrémités de la chaîne de traitement, des convertisseurs SGML↔objets permettent l'utilisation et le stockage des données sous des formats standard ; au niveau III, les formats MUC et LORIA sont acceptés. Il est d'autre part clair, en ce qui concerne les ressources, que le niveau de sortie ne peut excéder le niveau d'entrée plus l'annotation ou le traitement.

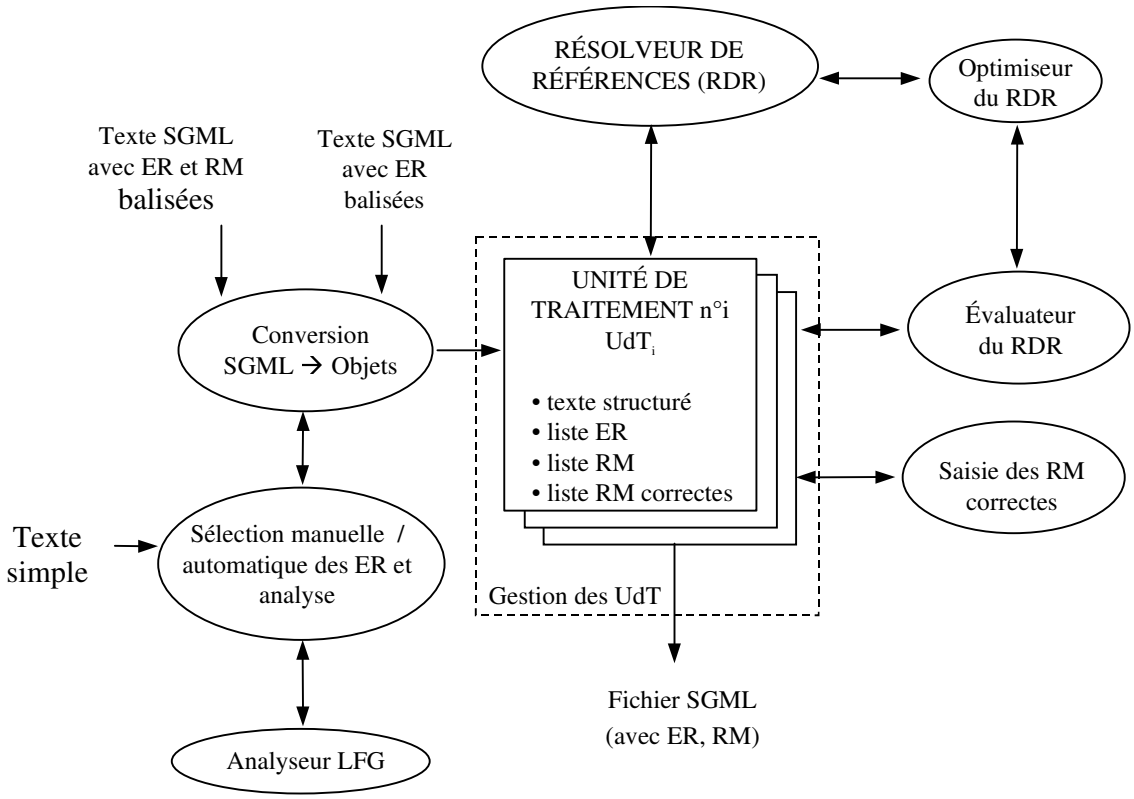

Figure VII-4. Architecture de l'Atelier de Traitement de la Référence (ATR).

D'autres architectures de traitement de la référence (au sens large) ont été proposées, voire distribuées via Internet, comme par exemple ALEMBIC (Day, Aberdeen, Hirschman, Kozierok, Robinson et Vilain 1997) ou GATE (Cunningham, Humphreys, Gaizauskas et Wilks 1997). Par rapport à celles-ci, notre atelier possède d'utiles convertisseurs de formats SGML, intègre le résolveur et les interfaces d'annotation, et implémente, dans le module d'évaluation, toutes les mesures qui ont été proposées jusqu'à présent (Chapitre IX).

VII.F.2 Le module d'optimisation

Nous avons implémenté un module qui modifie les paramètres numériques du résolveur (les paramètres d'activation ou bien la « mémoire ») pour faire augmenter le score du résolveur sur une unité de traitement donnée[13]. Cette optimisation se fait avec peu d'efforts grâce à l'intégration de nos outils, notamment l'évaluateur et le résolveur. Le module d'optimisation (interface en annexe) utilise la méthode du gradient, en considérant les scores comme un champ de potentiel ; la valeur initiale des paramètres est réglée empiriquement.

L'unité d'optimisation est le *cycle* : à chaque cycle, le programme choisit un paramètre au hasard, et l'incrémente puis le décrémente d'une certaine valeur

[13] Ce module est décrit plus en détail dans (Popescu-Belis 1998).

(paramétrable). Les deux scores obtenus (pour une mesure fixée à l'avance) sont comparés au score initial. En fonction du meilleur score, le paramètre est incrémenté, décrémenté, ou laissé inchangé ; si deux scores sont égaux, le programme tire au sort. Puis, un autre cycle commence, avec un nouveau paramètre au hasard.

Cette méthode d'optimisation garantit que le score est toujours amélioré, ou reste constant. Naturellement, elle peut aboutir à un maximum local, et non pas global. Toutefois, comme les résultats numériques vont le montrer (VII.G.4 ci-dessous), l'aspect du champ de potentiel est très « plat », ce qui rend la recherche très lente. De surcroît, chaque cycle nécessite trois résolutions et évaluations, ce qui ralentit encore le processus. L'amélioration obtenue peut donc paraître faible, mais elle ne nécessite aucun effort de la part du programmeur, une fois ces outils réalisés.

VII.G Résultats de notre système

Nous passons maintenant à la description et à l'analyse des résultats obtenus par notre système sur les textes décrits ci-dessus (VII.D.4, Table VII-2), mesurés à l'aide des méthodes d'évaluation (VII.E). Nous donnons d'abord les meilleurs résultats obtenus, i.e. dans la meilleure configuration du système (VII.G.1). Puis, nous examinons l'évolution des résultats en fonction des nombreux paramètres qui peuvent être modifiés dans le système. On étudie ainsi l'heuristique de compatibilité (définie au VII.C.3.iii, résultats au VII.G.2), puis la taille de la mémoire des RM (résultats au VII.G.3), et les paramètres d'activation (définis au VII.C.3.v, résultats au VII.G.4), avec des résultats sur l'optimisation de ces derniers. Enfin, nous proposons au VII.G.5.i une mesure de la pertinence des règles de compatibilité référentielle entre ER (décrites au VII.C.3.iv), et donnons les résultats obtenus (VII.G.5.ii) ; le même principe est utilisé pour l'étude de contraintes sur la (non)création de RM (VII.G.5.iii).

Une remarque valable pour l'ensemble de ces résultats est que nous ne donnons en général que deux chiffres significatifs (pourcentages entiers), puisque l'inexistence d'un accord parfait sur le balisage de la clé rend inutile une précision plus grande. Nous ne donnons pas les chiffres pour toutes les mesures, pour ne pas charger les tableaux ; le lecteur intéressé peut les trouver au IX.F.2, avec l'analyse des mesures.

VII.G.1 Meilleurs rappel et précision pour les différentes méthodes d'évaluation

Nos meilleurs scores pour diverses mesures sont donné sur la Table VII-3, et Table VII-4 pour les *f-mesures* associées. Les chiffres sont du même ordre que ceux obtenus par les systèmes présentés à MUC-6. Toutefois, étant donné que notre système bénéficie d'un ensemble d'ER identique à celui de la clé, ces chiffres reflètent une performance quelque peu inférieure. Cela n'est pas étonnant vu le faible niveau des connaissances et des outils linguistiques du système, ainsi que sa destination à des domaines non restreints.

%	MRS	MPS	BRS	BPS	CRS	CPS	XRS	XPS	D-erreur	Rcvt
VA	70	78	75	75	53	47	70	79	15-R	85
LPG.eq	62	77	50	57	43	36	41	65	17-R	73
LPG	70	88	37	52	43	43	35	61	14-R	66

Table VII-3. Résultats de notre système sur les trois textes (%).

Comme évoqué au VII.E.3.ii.a, notre mesure C (noyaux) est plus sévère que la mesure MUC. Par ailleurs, pour ces deux mesures, les résultats meilleurs sur LPG que sur LPG.eq montrent qu'un taux de coréférence important est un facteur d'indulgence : on peut estimer que la « qualité » du système ne change pas de LPG.eq à LPG. Les chiffres selon la mesure C reflètent mieux, à nos yeux, les performances du système.

%	MUC	B^3	C (noyaux)	XC (noy-ex)
VA	74	75	50	74
LPG.eq	69	53	39	50
LPG	78	43	43	44

Table VII-4. F-mesures *des résultats donnés sur la Table VII-3 (%).*

Les scores selon la mesure XC varient entre les textes (sur la Table VII-3) de façon opposée aux mesures MUC et C. Il en est de même pour les scores distributionnels, qui ont aussi tendance à diminuer de VA à LPG.eq à LPG. Les scores XC ne semblent donc pas être sensibles au taux de coréférence, ou plutôt y être sensibles dans le sens opposé. On peut en fait se demander si, bien que la « qualité » du système reste la même de LPG.eq à LPG, ses performances ne se dégradent pas à cause de l'accumulation d'erreurs. Nous laissons donc ouverte l'interrogation : à « qualité » constante du système, et avec une mesure « objective », les scores du système devraient-ils baisser sur des textes de plus en plus longs ?

VII.G.2 Comparaison des heuristiques de compatibilité

Nous avons montré au VII.C.3.iii que l'application des contraintes de sélection entre l'ER courante et l'ensemble des ER d'une RM pouvait être réalisée selon plusieurs heuristiques de compatibilité : H1 – compatibilité de l'ER courante avec la première des ER de la RM ; H2 – compatibilité avec toutes (non pronominales) ; H3 – compatibilité avec au moins une (non pronominale) ; H4 – compatibilité avec *X*%. Nous avons testé les trois premières sur notre système, tous les autres paramètres étant maintenus constants.

		H1 : *première*		**H2 :** *toutes*		**H3 :** *une*	
	%	R	P	R	P	R	P
VA	MUC	66	60	66	60	70	60
	C	52	44	52	44	56	39
	XC	62	73	63	73	60	69
LPG.eq	MUC	72	76	66	70	72	76
	C	57	34	40	35	57	34
	XC	40	54	38	54	40	54
LPG	MUC	80	85	77	83	80	85
	C	38	40	34	42	38	40
	XC	29	48	28	48	29	48

Table VII-5. Les scores pour les différentes heuristiques de sélection.

Les résultats obtenus (Table VII-5) montre que H3 obtient pour les trois essais les mêmes résultats que H1, qui sont meilleurs que ceux de H2. Ceci s'explique probablement par le fait que H2 est trop restrictive : imposer la compatibilité référentielle avec toutes les ER de la RM, alors même que le système n'est pas toujours capable de détecter celle-ci pour deux ER quelconques, revient à interdire de nombreux rattachements d'ER corrects.

Nous avons aussi expérimenté la variation de *X*, le coefficient intervenant dans H4 ; lorsque $X = 100\%$, on retrouve H2, et lorsque $X \rightarrow 0\%$, il existe une valeur qui permet de retrouver H3. *A priori*, cela ne rendrait pas impossible l'existence d'une valeur intermédiaire pour laquelle H4 soit à la fois meilleure que H2 et H3. Une expérience que nous avons réalisée prouve qu'il n'en est rien, et qu'on ne peut améliorer H3. Comment expliquer ce résultat ? D'une part, on peut penser qu'une valeur fixe de *X* ne conduit pas aux résultats corrects pour toutes les RM, car celles-ci ont des tailles très variées (entre une seule et des dizaines d'ER). D'autre part, la règle de compatibilité sémantique étant dans notre programme assez restrictive, la compatibilité entre l'ER courante et une seule des ER de la RM suffit pour confirmer la compatibilité avec la RM, ce qui fait que H3 reste une bonne heuristique.

Cela est confirmé par une autre expérience : nous avons restreint l'application des heuristiques à une partie du champ *RM.Liste(ER)* de chaque RM, partie intitulée *RM.ListeCaract(ER)* – les ER « caractéristiques » de la RM. Bien sûr, si on disposait d'une structure sémantique pour chaque RM, il serait inutile de déterminer un ensemble de désignations caractéristiques. Ici, nous pouvons régler la taille de cette *RM.ListeCaract(ER)* et son mode de construction : on peut (1) retenir les ER non pronominales les plus longues (toutes différentes), ou bien (2) les plus courtes. L'expérience montre que les résultats de (1) sont bien inférieurs à ceux de (2), ce qui signifie qu'il y a parmi les ER non pronominales les plus courtes des ER qui fonctionnent

comme des « noms standard » pour la RM, et qu'il faut absolument retenir dans *RM.ListeCaract(ER)*. Cette hypothèse est corroborée par le score raisonnable de H1, qui considère que la première référence à une entité se fait en général à l'aide d'un « nom standard ».

VII.G.3 Taille optimale de la « mémoire à court terme »

Nous avons étudié l'influence de la taille de la mémoire des RM accessibles aux références (VII.C.3.v), lorsque cette taille varie de 3 à 45 RM (mesurée toutes les 2 RM, cf. Figure VII-5). Lorsque cette mémoire est petite, des référents importants sont oubliés, donc le rappel est faible (à gauche sur la figure). Lorsque cette mémoire est trop grande, le système peut rattacher à tort l'ER courante à des référents erronés qui auraient dû être oubliés par un lecteur humain (à droite sur la figure), car le nombre de référents disponibles dépasse la sélectivité des heuristiques de compatibilité référentielle.

Il doit donc exister une capacité de la mémoire qui optimise la *f-mesure*, moyenne harmonique du rappel et de la précision. Sur la Figure VII-5, les lieux de *f-mesure* constante sont des hyperboles (on a représenté celle à 68%). Un résultat est d'autant meilleur qu'il est proche de l'angle supérieur droit du graphique. Les résultats montrent que l'optimum de la capacité mémoire se situe entre 15 et 20 ER, en moyenne sur les deux textes étudiés, et avec plusieurs jeux de paramètres. Ces valeurs permettent aussi d'accroître sensiblement la vitesse de traitement, par rapport à une mémoire illimitée.

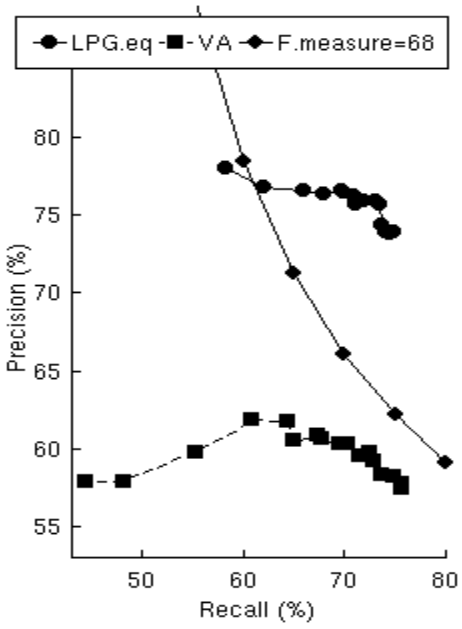

Figure VII-5. Influence de la capacité de la mémoire sur le rappel et la précision. Courbes pour VA, LPG.eq et « f-mesure=68% ». Les mesures commencent à 3 RM (à gauche) et vont jusqu'à 45 RM (à droite), par sauts de 2.

VII.G.4 Optimisation des paramètres d'activation

Nous avons expérimenté l'algorithme d'optimisation décrit au VII.F.2. Comme nous l'avions remarqué, les scores sont nécessairement améliorés, mais les jeux de paramètres correspondants n'ont en général pas d'interprétation évidente (ce qui est normal, vu la méthode de recherche). L'optimisation se déroule sur un texte donné, et à ce titre présente le risque d'être particulière à ce texte (le système « apprend le texte »). C'est pourquoi nous avons expérimenté avec deux textes longs, VA et LPG.eq ; leur longueur fait qu'il n'y a pas de lien visible entre la réponse correcte et le jeu de paramètres. Les résultats d'une optimisation ont été testés sur l'autre texte, comme nous allons le montrer.

Une première expérimentation a conduit sur VA à un gain en [*rappel* + *précision*] de 4,3% selon la mesure MUC. Sur LPG, une optimisation a fait gagner +2,5%, et une autre +5,9% (le processus n'est pas déterministe). L'algorithme est arrêté lorsque l'amélioration des scores stagne pendant un certain temps. Ce processus est toutefois très lent, et il n'a permis de gagner que 0.5% sur LPG (valeur non significative).

Dans une deuxième expérimentation, nous sommes partis d'un même jeu de paramètres J0, avec les scores donnés Table VII-6 pour différentes mesures de qualité. Nous avons alors réalisé deux optimisations de <*CRS*, *CPS*>, chacune de 2500 cycles, avec notre mesure noyaux C sur chacun des deux textes.

%	VA		LPG .eq		LPG	
MRS, MPS	71	55	77	75	86	87
CRS, CPS	**51**	**33**	**69**	**28**	70	31
XRS, XPS	55	64	39	48	28	39
D-erreur/Rcvt	27 P	78	32 P	63	37 P	51

Table VII-6. Scores pour le jeu de paramètres initial J0 : rappels, précisions et mesures distributionnelles, en pourcentages.

Le résultat de l'optimisation sur LPG.eq est donné sur la Table VII-7 ; le jeu de paramètres obtenus est appelé J1. On observe bien sûr une amélioration de *CRS* et de *CPS* (en gras), mais aussi des autres scores – les mesures sont donc covariantes, ce qui est un signe plus sûr d'amélioration.

%	VA		LPG .eq		LPG	
MRS, MPS	70	54	78	76	87	88
CRS, CPS	51	33	**70**	**30**	71	31
XRS, XPS	55	64	40	50	30	39
D-erreur, Rcvt	27 P	78	33 P	63	37 P	51

Table VII-7. Scores pour le jeu de paramètres J1 optimisé sur LPG.eq.

Le résultat de l'optimisation sur VA (jeu de paramètres J2) est donné sur la Table VII-8 ; à nouveau, les scores C mais aussi les autres ont été améliorés pour le texte VA.

%	VA		LPG .eq		LPG	
MRS, MPS	72	55	77	75	87	87
CRS, CPS	**52**	**34**	69	28	70	31
XRS, XPS	55	65	39	49	29	40
D-erreur, Rcvt	28 P	78	32 P	63	37 P	52

Table VII-8. Scores pour le jeu de paramètres J2 optimisé sur VA.

Un résultat bien plus intéressant est que les jeux de paramètres résultant de l'optimisation conduisent à de meilleurs résultats aussi sur l'*autre* texte, celui sur lequel ils n'ont pas été optimisés. Ainsi, J1 conduit à des scores meilleurs pour VA, et J2 pour LPG.eq, même si l'amélioration est moins grande que pour le texte sur lequel on a optimisé. La synthèse qualitative de toutes ces comparaisons est donnée sur la Table VII-9 (figurent entre chevrons les deux résultats prévisibles : <+>). On constate donc une amélioration généralisée des performances, y compris sur LPG, qui n'était pas intervenu directement dans l'optimisation.

	VA		LPG .eq		LPG	
	J1	J2	J1	J2	J1	J2
MRS+MPS	–	+	+	=	+	+
CRS+CPS	=	<+>	<+>	=	+	=
XRS+XPS	=	+	+	+	+	+
D-erreur+Rcvt	=	+	+	=	=	+

Table VII-9. Augmentation (+) ou diminution (–) des scores pour les deux jeux de paramètres (en moyenne sur le rappel et la précision ; <+> indique la valeur optimisée, donc prévisible).

Cette expérience montre donc sur un exemple que l'optimisation des paramètres n'est pas particulière au texte sur lequel on la réalise, et produit des jeux de paramètres meilleurs pour les trois textes dont nous disposons. On peut bien sûr relever plusieurs limitations : les gains obtenus ne sont pas très importants, car le potentiel des scores est assez plat (c'est une des raisons pour lesquelles nous avons cherché des scores plus sévères) ; la recherche est lente ; on ne peut sortir facilement des maxima locaux (les changements aléatoires sont inefficaces dans un champ de potentiel aussi plat).

VII.G.5 Mesure de la pertinence des différentes règles

VII.G.5.i Méthode

Une mesure de la contribution de chacune des connaissances ou règles d'un système au score final permettrait de vérifier si la pertinence pratique (mesurée) de chaque règle correspond bien à l'importance théorique qu'on lui attribue (estimée *a priori*). On pourrait alors concentrer les efforts sur les règles qui combinent une pertinence importante et un perfectionnement aisé.

Pour effectuer cette mesure, il est à nos yeux nécessaire de considérer un ensemble de règles ou connaissances qui soient utilisées alternativement par un système, c'est-à-dire indépendantes les unes des autres – il est plus difficile de mesurer l'importance relative de règles qui s'enchaînent. Il faut d'autre part que ces règles puissent être neutralisées ou supprimées sans que le système s'arrête. Ici, nous allons étudier d'abord les règles de compatibilité référentielle entre ER, puis des règles de création de RM ; mais auparavant, nous allons définir la méthode.

Si S est un score obtenu par un système sur un texte donné (rappel, précision ou autre), on souhaite idéalement exprimer la contribution de chaque règle comme une fraction de S. Formellement, si les règles sont notées R_i ($1 \le i \le N$), et leurs contributions à S sont notées $C(R_i)$, ces dernières doivent vérifier : $\sum_{1 \le i \le N} C(R_i) = S$.

On peut tenter deux propositions simples. D'abord, on peut considérer la baisse de S lorsque R_i seule est absente, en prenant pour mesure de l'utilité de R_i la valeur $C(R_i)$ égale à $S - S_{sauf}(R_i)$. Ou bien, on peut considérer comme valeur de $C(R_i)$ le score obtenu en gardant seulement R_i, score noté $S_{seul}(R_i)$. Prenons l'exemple suivant de scores, lorsqu'il y a trois règles :

Toutes	Sans R_1	Sans R_2	Sans R_3	R_1 seule	R_2 seule	R_3 seule
S	$S_{sauf}(R_1)$	$S_{sauf}(R_2)$	$S_{sauf}(R_3)$	$S_{seul}(R_1)$	$S_{seul}(R_2)$	$S_{seul}(R_3)$
50%	41%	41.6%	17.4%	16.4%	11%	38.8%

Table VII-10. Exemples de scores avec/sans certaines règles.

On pourrait donc estimer que la contribution de R_1 vaut 16.4/50 = 32.8%, celle de R_2 vaut 11/50 = 22%, celle de R_3 vaut 38.8/50 = 77.6% ; le problème est bien sûr que la somme de ces coefficients dépasse 100% (puisque la somme des scores quand on enlève R_1, ou R_2, ou R_3 dépasse le score total). On peut aussi dire que puisque la suppression de R_1 fait baisser le score de 9%, sa contribution est de 9/50 = 18%, et de même pour les deux autres (on trouve 16.8% et 65.2%) ; cette fois-ci la somme des scores vaut 100% … mais il ne s'agit que d'une coïncidence, comme on peut le vérifier sur les résultats ci-après. Ceux-ci montrent aussi que supprimer une règle ne fait pas toujours diminuer *tous* les scores.

Ce qu'il faut retenir de cet exemple, c'est que toutes les deux propositions de scores sont incohérentes : les règles n'étant pas indépendantes, leurs contributions ne peuvent être sommées linéairement. En revanche, si on regarde les deux triplets de scores obtenus, on s'aperçoit que leurs proportions sont semblables, et que ceci peut constituer un indice cohérent, dans ce cas, de la pertinence relative des trois règles (R_3 ayant sur l'exemple la plus grande importance).

Notre proposition est d'utiliser comme score une *f-mesure* (MUC ou C par exemple) et calculer pour chaque règle les scores précédents (Table VII-10), que nous notons (x_i, y_i) avec x_i pour le score sans R_i, y_i pour le score avec R_i seule. On ordonne alors les règles selon l'ordre standard (partiel) des couples (x, y) : $\{(x_1, y_1) < (x_2, y_2)\}$ *ssi* $\{x_1 < x_2$ et $y_1 < y_2\}$. Cet ordre est alors l'*ordre de pertinence* des règles, et les résultats expérimentaux ci-après suggèrent qu'en réalité il s'agit d'un ordre total. Il est en effet rationnel que les deux coefficients x_i et y_i soient covariants : si la suppression d'une règle entraîne une baisse importante du score, alors cette règle seule produira un bon score.

VII.G.5.ii Pertinence des règles de compatibilité référentielle

Nous avons étudié la pertinence des trois règles de compatibilité référentielle entre ER décrites au VII.C.3.iv : R_G (compatibilité en genre), R_N (compatibilité en nombre) and R_S (compatibilité sémantique). Pour ce faire, nous avons neutralisé tour à tour une, deux ou trois règles, et noté les scores du système selon trois *f-mesures* : MUC, C (noyaux) et XC (noyaux exclusifs) ; les résultats se trouvent Table VII-11 pour VA et Table VII-12 pour LPG. L'amplitude de la mesure C étant la plus grande, nous donnons aussi son rappel et précision. La première ligne montre les scores avec les trois règles, et les suivantes les différences lors des suppressions de règles : à savoir, pour un score S valant x%, la différence peut aller de $-x$% (chute du score à 0) jusqu'à $100-x$% (maximisation du score).

	R_G	R_N	R_S	MUC f-mesure	C rappel	C précision	C f-mesure	XC f-mesure
1	x	x	x	**65.5**	**53.4**	**50.0**	**51.6**	**70.5**
2		x	x	–0.5	–7.1	–9.0	–8.2	–5.0
3	x		x	–1.1	–5.6	–8.4	–7.2	–6.6
4	x	x		–7.9	+36.5	–32.6	–22.4	–49.4
5			x	–0.4	–5.3	–11.2	–8.7	–6.4
6	x			–7.3	+36.7	–33.6	–23.9	–50.4
7		x		–6.4	+44.0	–39.0	–31.9	–55.7
8				–6.6	+46.6	–40.2	–33.9	–60.6

Table VII-11. Variation des scores (lignes 2–8 différences avec la première) lorsque les règles de compatibilité sont alternativement supprimées, sur le texte VA.

On observe, sur les deux textes, que toute suppression fait diminuer les *f-mesures* : chaque règle est donc utile (!). On voit même que la plus mauvaise réponse (suppression des trois règles), bien qu'elle n'annule pas les *f-mesures*, fait baisser considérablement les résultats du système (sauf pour les mesures MUC, trop indulgentes), ce qui montre la pertinence de l'ensemble.

Il est naturel que le rappel (C) augmente dans plusieurs cas, puisque ces trois règles ont pour rôle de *limiter* le nombre de rattachements d'ER. Leur suppression entraîne l'augmentation de ces rattachements, parmi lesquels certains seront corrects. Cette augmentation est la plus notable pour le filtre sémantique R_S. Sa suppression entraîne aussi la baisse la plus significative de la précision et de la *f-mesure*, quelles que soient les règles présentes (R_G, R_N, ou les deux)[14]. Ces résultats désignent de façon cohérente R_S comme la règle de compatibilité la plus importante.

[14] Pour voir ceci, comparer la précision et la *f-mesure* des lignes 2, 3 et 5 avec celles des lignes 4, 6 and 7 ; ces dernières sont trois fois plus grandes que les premières.

	R_G	R_N	R_S	MUC f-mesure	C rappel	C précision	C f-mesure	XC f-mesure
1	x	x	x	**78.4**	**41.4**	**44.5**	**42.9**	**41.5**
2		x	x	+0.3	+4.1	–1.9	+1.1	–0.4
3	x		x	+0.8	+0.3	–1.8	–0.7	–0.9
4	x	x		+8.5	+40.6	–19.6	–4.8	–14.8
5			x	+1.6	+4.9	–3.2	+0.8	–1.2
6	x			+9.2	+42.1	–20.8	–6.0	–16.0
7		x		+9.4	+49.2	–24.9	–10.6	–20.0
8				+13.9	+58.6	–26.8	–12.8	–23.8

Table VII-12. Variation des scores (lignes 2–8 différence avec la première) lorsque les règles de compatibilité sont alternativement supprimées, sur le texte LPG.

Parmi les deux autres règles, il apparaît à nouveau clairement, quoique de façon moins nette, que l'accord en genre R_G est plus important que l'accord en nombre R_N. En effet, la suppression de R_G induit toujours une baisse plus importante de la *f-mesure* (et de la précision) que la suppression de R_N, et cela que R_S soit présente (ligne 2 vs. ligne 3) ou absente (ligne 7 vs. ligne 6). Par ailleurs, l'augmentation du rappel est aussi plus importante lorsque R_G est supprimée. Cette plus grande importance de R_G est aisée à comprendre : le nombre d'ER plurielles coréférentes est assez réduit, par rapport au nombre de situations où la considération du genre est pertinente (par exemple les pronoms au singulier).

VII.G.5.iii Pertinence des règles pour les GN définis/indéfinis

Nous avons appliqué les mêmes considérations à un autre ensemble de règles, à savoir les liens (déjà longuement discutés) entre le caractère défini ou non d'un GN et la création ou non d'une RM. Nous avons donc implémenté les deux règles :

- ER est un GN indéfini ⇒ créer une nouvelle RM ;
- ER est un GN défini ⇒ rattacher l'ER à une RM existante.

Nous avons testé les quatre possibilités où aucune, une seule ou les deux règles sont actives. Les résultats sont donnés dans la Table VII-13 (VA) et la Table VII-14 (LPG), selon le même principe de différences que les tables précédentes.

Créer une RM pour un GN indéfini	Associer un GN défini à une RM	MUC	C			XC
		f-mesure	rappel	précision	f-mesure	f-mesure
optionnel	optionnel	**71.9**	**47.4**	**49.0**	**48.2**	**73.8**
obligatoire	optionnel	–1.0	–4.5	+1.0	–2.0	+1.0
optionnel	obligatoire	–20.3	–7.5	–21.9	–16.0	–21.3
obligatoire	obligatoire	–21.7	–9.4	–23.3	–17.6	–20.4

Table VII-13. Variation des scores (lignes 2, 3, 4 par rapport à la première) pour les règles de création de RM, sur le texte VA.

Créer une RM pour un GN indéfini	Associer un GN défini à une RM	MUC	C			XC
		f-mesure	rappel	précision	f-mesure	f-mesure
optionnel	optionnel	**80.0**	**46.4**	**41.3**	**43.7**	**40.3**
obligatoire	optionnel	–7.3	–1.3	+4.5	+1.8	+3.2
optionnel	obligatoire	–5.8	–1.8	–2.4	–2.1	–0.4
obligatoire	obligatoire	–11.9	–2.4	–1.8	–2.1	–0.2

Table VII-14. Variation des scores (lignes 2, 3, 4 par rapport à la première) pour les règles de création de RM, sur le texte LPG.

Il apparaît assez clairement qu'aucune des deux règles, pas plus que leur conjonction, ne contribue à améliorer les scores – ceux-ci baissent dans la majorité des cas, pour les trois mesures. La première règle semble améliorer certains scores, mais ces deux exemples ne permettent pas de conclure quant à son (in)utilité. Plusieurs raisons peuvent être invoquées, et d'abord des raisons techniques, mettant en cause l'expérimentation. Ainsi, les deux textes étudiés sont de longs récits, d'un seul tenant, donc l'introduction de nouveau référents n'est pas aussi fréquente que dans une suite de dépêches (par exemple). En outre, l'imprécision de l'analyseur syntaxique utilisé fait que certains traits 'défini' ou 'indéfini' lui échappent, ce qui biaise les résultats. Des outils plus précis sont sans doute nécessaires pour évaluer statistiquement l'utilité des deux règles énoncées, ou de variantes plus fines[15].

Mais sur un plan plus conceptuel, il existe aussi des arguments linguistiques pour mettre en doute l'utilité des deux règles énoncées. Nous avons vu au Chapitre V que les théories logicistes de la référence, reprenant les idées de Frege ou de Russell, défendent dans leurs grandes lignes ces deux règles ; les travaux de Karttunen (cf. V.B.2.iv) visent entre autres à les raffiner. Mais nous avons surtout montré par la suite que les théories

[15] Notre but ici est surtout d'exemplifier la technique d'évaluation de ces règles.

fonctionnalistes (VI.B), et nos propres expériences (**VI.C**) suggèrent que les articles définis en anglais et en français sont loin d'avoir un comportement aussi « cartésien ».

VII.G.5.iv Pertinence des paramètres numériques

La méthode d'évaluation décrite dans cette section ne semble pas applicable aux coefficients d'activation des RM, puisque leur suppression alternée produit des effets très variables en fonction des *valeurs* des paramètres restants. Les paramètres numériques ne sont pas du type présent/absent, mais décrivent un ensemble de valeurs : ce qui est pertinent pour ces paramètres, c'est l'optimisation de ces valeurs (VII.G.4).

Si tous les paramètres d'activation sont mis à 0, ce qui revient à supprimer le mécanisme d'activation et d'oubli, alors les scores diminuent en moyenne de 50%. Cela confirme l'utilité de ce mécanisme, dans notre implémentation munie de peu de connaissances – contrairement aux résultats de Azzam et al. (1998), pour un système muni de connaissances sémantiques élaborées. En ce qui concerne la taille de la mémoire des RM, nous avons vu (VII.G.3) que sa valeur optimale se situait entre 15 et 20 RM, et la « suppression » de ce paramètre (correspondant à une mémoire illimitée) conduit à des scores inférieurs.

VII.H Conclusions

VII.H.1 Conclusion sur notre système

Nous avons parcouru dans ce chapitre le chemin menant de la conception d'un modèle réaliste et implémentable de la référence dans les textes, jusqu'à son implémentation et à son évaluation – et cela en passant par la définition d'un algorithme de résolution (robuste et utilisant des connaissances simples), par un protocole et des exemples de construction de ressources (conventions d'annotation et interfaces), par un système d'intégration des outils (l'ATR) et par des mesures d'évaluation (détaillées dans la troisième partie de cette thèse).

Le modèle proposé conduit à des résultats comparables à ceux des autres systèmes testés dans des conditions semblables. Avec peu de ressources disponibles, nous avons montré que le paradigme des RM, ou plus exactement l'ensemble des techniques de résolution employées, conduit à de bien meilleurs résultats que d'autres techniques proposées, notamment celles qui sont fondées sur la simple coréférence entre ER, ou qui ne tiennent pas compte de l'activation des entités.

Les développements à venir devront être guidés par un classement fréquentiel des phénomènes linguistiques qu'il faut traiter, afin de commencer par ceux qui ont un rapport fréquence/complexité avantageux. Dans l'immédiat, l'amélioration des connaissances sémantiques du système est une priorité mise en avant par l'importance de la compatibilité sémantique (VII.G.5.ii ci-dessus). La construction de ressources pour la langue française permettrait d'autre part une évaluation plus nuancée.

VII.H.2 Avantages et limites de notre approche

Le modèle simplifié des représentations mentales que nous avons proposé nous a permis de concevoir et d'implémenter un système robuste de résolution de la référence. Ce modèle permet de formuler de façon cohérente des questions aussi diverses que l'annotation ou l'évaluation. Les résultats concrets obtenus sont encourageants, mais le système demande à être enrichi avec des connaissances plus élaborées. Malgré la difficulté de trouver des ressources suffisamment étendues, compatibles avec la robustesse du résolveur, nous avons incorporé dans le modèle les ressources qui seront le plus probablement disponibles dans un avenir proche.

Dans une perspective plus générale, on peut aussi observer que le modèle théorique proposé dans le projet CERVICAL permet d'enrichir encore notre modèle, par l'ajout de plusieurs « entrées » à nos RM. Or, il est notable que ce projet, bien que défendant l'utilité d'une « entrée visuelle » (par exemple), n'aborde pas les façons concrètes d'instancier et d'utiliser une telle entrée (un tel niveau de détail ne faisant pas partie de ses buts).

Une résolution réaliste de la référence demande non seulement la construction (dans un langage symbolique) de représentations mentales, mais elle requiert aussi la transformation du programme de résolution en un véritable *agent* obéissant aux propositions de la « pragmatique simulée ». À savoir, pour passer d'un *traitement* de la référence à une *compréhension* de la référence, le programme doit être capable d'un ancrage perceptif de ses représentations, qui porteront sur des entités de son environnement. À ce titre, le modèle structurel de la capacité référentielle d'un agent, décrit au chapitre précédent, montre la direction des futurs développements.