Extraction d'Information et modélisation de connaissances à partir de Notes de Communication Orale

Fabrice Even

École Centrale de Nantes **Université de Nantes** **École des Mines de Nantes**

ÉCOLE DOCTORALE STIM

« SCIENCES ET TECHNOLOGIE DE L'INFORMATION ET DES MATERIAUX »

Année 2005

Extraction d'Information et modélisation de connaissances à partir de Notes de Communication Orale

---

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE NANTES

Discipline : INFORMATIQUE

présentée et soutenue publiquement par

Fabrice EVEN

le 5 octobre 2005

à l'UFR Sciences et Techniques, Université de Nantes

devant le jury ci-dessous

Président :	Alexandre DIKOVSKY, Professeur des Universités	LINA, Université de Nantes
Rapporteurs :	Pierre ZWEIGENBAUM, Professeur des Universités	INSERM, Hôpitaux de Paris
	François ROUSSELOT, Maître de conférences	LIIA, INSA Strasbourg
Examinateurs :	Noureddine MOUADDIB, Professeur des Universités	LINA, Université de Nantes
	Chantal ENGUEHARD, Maître de conférences	LINA, Université de Nantes
	Pascal MUCKENHIRN	Crédit Mutuel LACO

Directeur de thèse : Professeur Noureddine MOUADDIB
Co-encadrante : Maître de conférences Chantal ENGUEHARD
Laboratoire : Laboratoire d'Informatique de Nantes Atlantique (LINA) CNRS-FRE 2729

N° ED 366-210

EXTRACTION D'INFORMATION ET MODELISATION DE CONNAISSANCES A PARTIR DE NOTES DE COMMUNICATION ORALE

Information Extraction and knowledge modelling from oral communication notes

Fabrice EVEN

favet neptunus eunti

Université de Nantes

Fabrice Even
Extraction d'Information et modélisation de connaissances à partir de Notes de Communication Orale
xviii+230.

La rivière coule sans jamais s'arrêter, pourtant les millions de gouttes d'eau qui la composent ne sont jamais les mêmes. Toute chose passe.
— Takeo KUMAGAMI.

Résumé

Malgré l'essor de l'Extraction d'Information et le développement de nombreuses applications dédiées lors de ces vingt dernières années, cette tâche rencontre des problèmes lorsqu'elle est réalisée sur des textes atypiques comme des Notes de Communication Orale.
Les Notes de Communication Orale sont des textes issus de prises de notes réalisées lors d'une communication orale (entretien, réunion, exposé, etc.) et dont le but est de synthétiser le contenu informatif de la communication. Leurs contraintes de rédaction (rapidité et limitation de la quantité d'écrits) sont à l'origine de particularités linguistiques auxquelles sont mal adaptées les méthodes classiques de Traitement Automatique des Langues et d'Extraction d'Information. Aussi, bien qu'elles soient riches en informations, elles ne sont pas exploitées par les systèmes extrayant des informations à partir de textes.
Dans cette thèse, nous proposons une méthode d'extraction adaptée aux Notes de Communication Orale. Cette méthode, nommée MEGET, est fondée sur une ontologie modélisant les connaissances contenues dans les textes et intéressantes du point de vue des informations recherchées (« ontologie d'extraction »). Cette ontologie est construite en unifiant une « ontologie des besoins », décrivant les informations à extraire, avec une « ontologie des termes », conceptualisant les termes du corpus à traiter liés avec ces informations. L'ontologie des termes est élaborée à partir d'une terminologie extraite des textes et enrichie par des termes issus de documents spécialisés. L'ontologie d'extraction est représentée par un ensemble de règles formelles qui sont fournies comme base de connaissance au système d'extraction SYGET. Ce système procède d'abord à un étiquetage des instances des éléments de l'ontologie d'extraction présentes dans les textes, puis extrait les informations recherchées. Cette approche est validée sur plusieurs corpus.

Mots-clés : Extraction d'Information, Note de Communication Orale, Traitement Automatique des Langues Naturelles, Ontologie, Modélisation, Terminologie

Abstract

In spite of the rise of Information Extraction and the development of many applications in the last twenty years, this task encounters problems when it is carried out on atypical texts such as oral communication notes.
Oral communication notes are texts which are the result of an oral communication (meeting, talk, etc.) and they aim to synthesize the informative contents of the communication. These constraints of drafting (speed and limited amount of writing) lead to linguistic characteristics which the traditional methods of Natural Language Processing and Information Extraction are badly adapted to. Although they are rich in information, they are not exploited by systems which extract information from texts.
In this thesis, we propose an extraction method adapted to oral communication notes. This method, called MEGET, is based on an ontology which depends on the information to be extracted ("extraction ontology"). This ontology is obtained by the unification of an "ontology of needs", which describe the information to be found, with an "ontology of terms" which conceptualize the terms of the corpus which are related to the required information. The ontology of terms is elaborated from terminology extracted from texts and enriched by terms found in specialized documents. The extraction ontology is formalized by a set of rules which are provided as a knowledge base for the extraction system SYGET. This system (1) carries out a labelling of each instance of every element of the extraction ontology and (2) extracts the information. This approach is validated in several corpora.

Keywords: Information Extraction, Oral Communication Note, Natural Language Processing, Ontology, Modelling, Terminology

Remerciements

Je tiens en premier lieu à remercier Chantal Enguehard qui a encadré mon travail de recherche durant cette thèse. Je lui exprime ma sincère gratitude pour son implication, son aide, son écoute, ses remarques et ses critiques qui m'ont toujours permis d'avancer.

Je remercie chaleureusement Noureddine Mouaddib et Pascal Muckenhirn pour leur aide et leur soutien tout au long de ces années de doctorat.

Je remercie grandement François Rousselot et Pierre Zweigenbaum pour avoir accepté d'être rapporteurs de ma thèse. Merci pour vos remarques et vos commentaires pertinents. Je remercie également vivement Alexandre Dikovsky pour m'avoir fait l'honneur de présider mon jury.

Un grand merci à Nordine Fourour et Benjamin Habegger avec qui j'ai eu la joie de partager le bureau 212 du LINA ainsi que beaucoup d'autres choses. Merci également à Lorraine Goeuriot pour m'avoir supporté dans ce même bureau lors des derniers moments de ma thèse.

Merci à toutes les personnes qui ont participé, de près ou de loin, à mes recherches et à l'élaboration de cette thèse. Je pense particulièrement aux membres de l'équipe TALN du LINA ainsi qu'à tous ceux, au laboratoire ou ailleurs, avec lesquels j'ai pu échanger avis, idées et conseils.

Merci aux étudiants, aux enseignants et aux autres personnels de la Faculté des Sciences et Techniques de Nantes et de l'École Polytechnique de l'Université de Nantes, avec lesquels j'ai pris beaucoup de plaisir à effectuer des enseignements.

Merci à LOGIN et à tous ses membres, passés ou présents, pour leur sympathie et leur disponibilité. Un clin d'œil particulier à ses présidents successifs qui ont su insuffler dynamisme et convivialité à cette association : bravo à Gaëtan, Gwen, Erwan, Sandra et Anthony.

Merci à ma famille et particulièrement à mes parents Jean-Pierre et Martine, mon frère Arnaud et ma sœur Justine pour leur affection et leurs encouragements.

Merci enfin à tous mes amis qui, à l'université ou en dehors, ont grandement contribué par leur présence, leur aide et leur appui, à l'accomplissement de cette thèse. Merci à Manu, Alexandra, Jérôme, Franck, Estelle, Lucas, Brice, Arnaud, Dallas, Sylvain, Élodie, Cédric, David, Mitch, Greg, Solène, Ghim, Fred, Anne-Gaëlle, Adrien, Solenne, Antoine, Gerson, Jim, Guillaume, Chloé, Éric, Charlotte, Pierre, Laura, Gilles, Éloïse, Jen, Marco, Morgan, Florence, Seb, Jérémie, Sophie, Ben, Alizée, Céline, Raphaëlle, Vincent, et tous ceux qui, même s'ils ne sont pas cités ici, se reconnaitront. Je vous dois beaucoup.

Table des matières

Table des figures

Table des exemples

Introduction

Cadre de la thèse

Cette thèse s'inscrit dans le domaine du *Traitement Automatique des Langues Naturelles* (*TALN*) et plus précisément dans celui de l'*Extraction d'Information* [Piazenza 1997]. Le but des travaux en Extraction d'Information est de développer des méthodes et des outils visant à extraire automatiquement des informations à partir de textes écrits en langue naturelle. Il s'agit d'analyser des documents textuels afin de collecter et de structurer des informations précises définies en amont. De manière générale, les types d'informations recherchées sont décrits formellement à travers des formulaires dits d'extraction. Cette technologie hérite des travaux en structuration puis en compréhension de textes. Elle a acquis sa maturité lors des années 1990 au cours desquelles ont émergé les premiers véritables systèmes d'Extraction d'Information. Des systèmes plus efficaces ont ensuite été développés en se fondant principalement sur des méthodes d'analyse linguistique et/ou d'apprentissage utilisant des outils et des techniques issues des recherches en TALN.

L'Extraction d'Information est une technologie récente mais qui cherche à répondre à un besoin très ancien : acquérir de la connaissance à partir de textes. Cette nécessité s'est accrue ces vingt dernières années avec l'essor considérable de la masse de documents disponibles au format électronique (Internet, courrier et documentation électronique) qu'il faut gérer afin d'extraire ou de filtrer les informations pertinentes parmi toutes celles contenues dans ces documents [Minel 2002]. Comme la Recherche d'Information, le résumé automatique ou les systèmes de Question-Réponse, l'Extraction d'Information a l'ambition de répondre à ce défi, d'où le développement de nombreuses applications destinées à des institutions ou au monde des affaires et/ou de l'industrie.

Problématique

Les différents travaux en Extraction d'Information et plus généralement en TALN s'intéressent quasi-exclusivement à des textes rédigés en conformité avec les normes d'écriture de leur langue. Ces normes correspondent à l'ensemble des règles ou prescriptions syntaxiques et lexicales fixées pour une langue et définissent des conventions de rédaction. Les systèmes d'Extraction d'Information existants traitent des corpus composés de textes issus de la presse (journaux, revues économiques, dépêches), de la littérature (livres, essais, actes de publications scientifiques) ou de documents officiels d'institutions ou d'entreprises (rapports d'expertise, bilans financiers), c'est-à-dire des textes qui *a priori* sont correctement écrits d'un point de vue grammatical et qui comportent peu de fautes.

Néanmoins, tous les textes ne respectent pas les normes usuelles d'écriture. Nombreux sont ceux qui s'écartent de manière plus ou moins importante des standards habituels de la langue. Il s'agit par exemple de textes rédigés en employant des règles d'écriture propres, éloignées des règles standards (comme les petites annonces, les SMS) ou de textes rédigés en utilisant des règles de rédaction de façon altérée, ce qui se traduit par la présence d'une grande quantité de fautes lexicales et/ou syntaxiques (textes issus de prise de notes, transcriptions de l'oral, messagerie électronique). Ces textes sont particulièrement courants au sein de corpus issus du monde économique ou industriel. Alors qu'ils véhiculent des informations intéressantes, le TALN voue peu d'effort à leur exploitation. Leurs spécificités sont rarement prises en compte dans les outils de traitement automatique fondés sur l'hypothèse que les textes à analyser suivent des règles bien établies.

Parmi la variété d'écrits s'écartant des normes usuelles d'écritures, nous nous sommes intéressé dans cette thèse aux *Notes de Communication Orale*. Nous regroupons sous cette dénomination les textes issus d'une prise de notes réalisée lors d'une communication orale (entretien, réunion, exposé, etc.). Il s'agit de textes rédigés rapidement et dont le but est de synthétiser le contenu informatif de la communication. Les contraintes de rédaction de ce type de textes sont à l'origine de particularités linguistiques qui posent problème aux méthodes usuelles de TALN et d'Extraction d'Information.

L'objectif de nos travaux est d'apporter des solutions au problème de l'Extraction d'Information à partir de Notes de Communication Orale. Dans ce but, nous avons d'abord dégagé et examiné l'ensemble des caractéristiques linguistiques de ce type de texte. Ensuite, nous avons cherché à comprendre et à analyser les raisons de l'inadéquation des techniques existantes d'Extraction d'Information vis-à-vis de tels textes afin d'en extraire des pistes de réflexion pour l'élaboration d'une méthode adaptée à leurs caractéristiques. La méthode d'extraction **MEGET** (Méthode Générique d'extraction d'Information à partir de Textes) que nous avons définie est le résultat de cette démarche.

Contexte

Cette thèse a été réalisée en partenariat avec le *Crédit Mutuel LACO* (*Crédit Mutuel Loire-Atlantique Centre-Ouest*), organisme bancaire disposant de très importantes quantités de Notes de Communication Orale qui s'avèrent très riches en informations utilisables à des fins commerciales. Nous avons pu disposer de ces corpus pour fonder nos analyses et nos expérimentations. Nous avons également pu évaluer tout au long de cette thèse les résultats de nos recherches avec des experts et des analystes bien au fait du domaine et des corpus à traiter. Ces évaluations ont permis une évolution continue de la méthode par l'amélioration des solutions mises en œuvre et l'exploration de nouvelles voies dont certaines ont abouti au développement de nouvelles solutions.

Organisation du document

Ce manuscrit se compose de deux parties.

Dans ***la première partie***, nous confrontons les travaux en Extraction d'Information avec les Notes de Communication Orale. Le **chapitre 1** présente le domaine de l'Extraction d'Information. Nous commençons par le définir avant de le placer dans son contexte en le situant historiquement et vis-à-vis d'autres domaines visant à traiter les informations présentes dans les textes. Dans le **chapitre 2**, nous commençons par une présentation générale des textes que nous qualifions de standards et de non-standards. Ensuite nous nous intéressons en détail aux Notes de Communication Orale. Nous définissons ce type de textes, décrivons ses caractéristiques linguistiques et étudions les problèmes qu'il pose aux systèmes usuels d'Extraction d'Information. Le **chapitre 3** aborde une réflexion fondée sur l'idée que modéliser le type d'information peut apparaître comme une solution pour en extraire des instances rencontrées dans les textes, et ce quelles que soient les particularités linguistiques des textes à traiter. Nous présentons les ontologies, un mode de représentation et de modélisation des connaissances particulièrement bien adapté à une utilisation dans une application informatique. Les principaux systèmes d'Extraction d'Information utilisant des ontologies sont ensuite passés en revue et confrontés avec les Notes de Communications Orale. Nous terminons ce chapitre en synthétisant les réflexions et analyses précédentes et aboutissons à l'élaboration d'une méthode d'extraction d'information fondée sur une ontologie et adaptée à ce type de textes. Le **chapitre 4** discute la notion de terme, et confronte les différentes techniques d'extraction de terminologie avec les Notes de Communication Orale afin d'évaluer la meilleure solution pour repérer dans les textes les manifestations linguistiques d'un ensemble de concepts.

La ***deuxième partie*** présente la méthode d'extraction **MEGET** (Méthode Générique d'Extraction d'information à partir de Textes). Cette méthode associe une ontologie modélisant les connaissances recherchées dans le corpus et représentée formellement par un ensemble de règles (*ontologie d'extraction*), avec un système automatique réalisant l'extraction des informations. Ce système, appelé **SYGET** (Système Générique d'Extraction d'information à partir de Textes), repère dans le texte les instances des concepts de l'ontologie. Le **chapitre 5** détaille une méthode pour élaborer une ontologie capable de modéliser les informations à rechercher dans les corpus, à travers la définition d'un modèle conceptuel fondé sur des prédicats (*ontologie des besoins*). Le **chapitre 6** présente une étude termino-ontologique réalisée afin de relier les concepts de l'ontologie des besoins avec les termes présents dans les textes à traiter. Cette étude aboutit à l'élaboration d'une *ontologie des termes* (ontologie issue des termes) qui sera unifiée à l'ontologie des besoins, le résultat de cette unification formant l'ontologie d'extraction. Nous décrivons les différentes étapes de la construction de l'ontologie des termes ainsi que la phase d'unification. Le **chapitre 7** décrit en détail le fonctionnement du système **SYGET**. Le dernier chapitre de cette thèse (**chapitre 8**) présente les résultats de plusieurs expérimentations.

Partie I

Extraire de l'information de Notes de Communication Orale

Nous sommes une espèce passionnée par la recherche mais qui a peur de découvrir. Nous répondons à nos peurs par nos croyances, un peu comme ces anciens marins qui refusaient l'idée du voyage, convaincus que chargés de leurs certitudes le monde s'achevait en un abîme sans fin.
— Marc Levy, Une prochaine fois.

Chapitre 1

L'Extraction d'Information : définitions et objectifs

Présentation

Ce chapitre présente le domaine de l'Extraction d'Information dans lequel se situent les travaux de recherche exposés dans ce manuscrit. Le domaine est d'abord défini (section 1.1), replacé dans un contexte historique et mis en perspective vis-à-vis des autres domaines de recherche en informatique cherchant à collecter de l'information à partir de textes en langue naturelle (section 1.2). Nous présentons ensuite les évolutions des recherches en Extraction d'Information depuis leurs origines jusqu'à la fin du 20$^{\text{ème}}$ siècle (section 1.3).

1.1 Définition

L'Extraction d'Information ou EI (en anglais, *Information Extraction* ou *IE*) désigne une technologie récente qui vise à extraire et à structurer automatiquement un ensemble d'informations précises apparaissant dans un ou plusieurs documents textuels écrits en langue naturelle.

Ces informations sont destinées à créer ou alimenter un entrepôt de données (appelé aussi banque de données) [Piazenza 1997]. La tâche d'extraction est réalisée grâce au remplissage de formulaires prédéfinis (*template*).

Ces formulaires, dits formulaires d'extraction, sont définis dans le but de représenter la connaissance à rechercher par une structure déterminée *a priori*. Ils décrivent un ensemble d'entités, les relations entre celles-ci et les événements impliquant ces entités [Yangarber & al. 2000]. Par exemple, un formulaire concernant des accidents de la route devra spécifier

des champs comme « Lieu de l'accident », « Nombre de victimes », « Identité des victimes » ou encore « Cause de l'accident ».

Les informations extraites par un système d'Extraction d'Information peuvent être consultées par des utilisateurs humains (par exemple via la génération de rapports d'événements), être utilisées pour la génération de résumés (dans la même langue ou dans une langue différente) ou, dans la plupart des cas, alimenter une base de données afin d'être analysées plus tard (interrogation par requêtes ou fouille de données[1]).

Un exemple d'extraction de faits de guerre à partir d'articles de journaux est présenté dans l'exemple 1.1.

Exemple 1.1 (Extraction d'Information sur un extrait du journal *Libération*)

Texte :

```
Libération - lundi 27 octobre 2003 -  Bagdad envoyé spécial -- Réveil
agité hier à 6 heures du matin pour Paul Wolfowitz, le numéro 2 du
Pentagone, qui passait la nuit à Bagdad dans l'hôtel Al-Rashid,
transformé en bunker par les forces d'occupation américaines. Au moins
six roquettes Katioucha, tirées depuis une remorque stationnée à 400
mètres de là, ont atteint la façade de l'hôtel de luxe. L'attaque a tué
un soldat américain et blessé quinze autres personnes, en majorité des
Américains.
```

Formulaire rempli :

Événement : attaque
Nature : tir de roquettes
Date : 27 octobre 2003
Lieu : hôtel Al-Rashid, Bagdad, Irak
Cible : Paul Wolfowitz
Victimes : **Mort** : un soldat américain
Blessés : quinze personnes, en majorité des Américains

À partir du formulaire précédent, il est possible de générer un résumé du texte correspondant (cas 1) ou d'alimenter une base de données (cas 2).

Cas 1 : génération de résumé

```
Une attaque au tir de roquettes a eu lieu le 27 octobre 2003 à l'hôtel
Al-Rashid à Bagdad. Elle visait Paul Wolfowitz, a tué un soldat
américain et blessé quinze personnes, en majorité des Américains.
```

Cas 2 : alimentation d'une base de données

Table ATTAQUE					
Type	**Cible**	**Date**	**Lieu**	**Victimes**	
				Mort	Blessé
Tir de roquettes	Paul Wolfowitz	27 octobre 2003	hôtel Al-Rashid, Bagdad, Irak	1	15

[1] Data Mining

1.2 Contexte

1.2.1 Un besoin ancien et essentiel

L'Extraction d'Information est désormais un sujet de recherche important dans le domaine du Traitement Automatique des Langues Naturelles. Elle connaît ces dernières années un intérêt grandissant car elle répond à un besoin devenu incontournable dans la société de l'information.

Il faut souligner que la collecte d'informations dans des textes est une activité qui remonte à l'Antiquité. Depuis que l'écriture existe, l'humanité s'est penchée sur les textes pour y trouver des réponses à ses questions, a étudié les écrits pour acquérir des connaissances. Cette quête de savoir a connu ces dernières décennies un essor considérable avec le passage à la civilisation de l'information dont une des principales conséquences est la production en masse de documents textuels sous format électronique. Ce phénomène est encore plus notable ces dernières années avec le développement d'Internet et des communications par courriers électroniques. L'augmentation de la quantité de textes électroniques est renforcée par l'apparition de capacités de stockage de plus en plus importantes.

1.2.1.1 Enjeux

Dans la plupart des domaines, qu'il s'agisse de l'économie, de la société ou de la sécurité, obtenir et traiter régulièrement des informations est devenu une nécessité [Wilks 1997], notamment afin de s'appuyer sur des bases solides lors des prises de décision.

Dans le domaine de l'économie, la collecte d'information est un enjeu essentiel. Les entreprises ont en permanence besoin d'informations fiables et pertinentes sur les marchés ainsi que sur leurs concurrents afin d'élaborer les stratégies leur permettant d'améliorer leurs résultats et de gagner des parts de marché[2]. Pour répondre à ce besoin, les acteurs économiques se tournent vers les documents issus de la presse, et principalement de la presse économique. Ce processus concerne particulièrement le monde de la finance dans lequel il est nécessaire de connaître au jour le jour les fluctuations au sein des différents secteurs de l'économie. Les prises de décision s'appuient sur des événements particuliers extraits de l'étude de très importantes quantités de textes. Par exemple, en Grande-Bretagne, la banque *Lloyds* emploie des centaines de personnes pour chercher quotidiennement dans des journaux du monde entier les naufrages de bateaux à travers le globe dans le cadre de son activité d'assureur.

Acquérir de l'information est également un enjeu au niveau sociologique, notamment pour les acteurs politiques. L'analyse de documents traitant d'une société amène à discerner et comprendre les comportements des différentes composantes d'une population, les multiples problèmes de la société et les opinions publiques. Ce type d'analyse permet de

[2] Veille Informationnelle : utilisation de moyens technologiques pour connaître les éléments et les mouvements stratégiques et opérationnels de l'environnement des organisations ou des entreprises.

définir et de proposer des politiques répondant à ces problèmes[3] ou de trouver les moyens de faire comprendre et accepter des mesures à une population.

Dans les secteurs de la défense ou de la sécurité, la collecte d'information a toujours été au cœur des services de renseignements militaires ou policiers. Elle est essentielle dans la lutte contre le terrorisme afin de déceler les prémisses d'actions terroristes. Au niveau militaro-politique, elle est utile en temps de paix pour découvrir les germes des futurs conflits, et en temps de guerre pour déceler certains faits et gestes ennemis afin de prévoir les stratégies militaires à mettre en place. Pour remplir ces objectifs, les services de renseignement se focalisent d'une part sur l'étude de documents traitant de sujets policiers ou militaires (dans la presse par exemple) et d'autre part sur l'analyse de textes relatant des correspondances (courriers papiers ou électroniques, transcriptions d'écoutes téléphoniques) ou de conversations (issues par exemple de l'espionnage d'individus au moyen de microphones).

1.2.1.2 Évolution de la tâche d'extraction

Les textes en langue naturelle véhiculent une grande quantité d'informations. Pour pouvoir analyser et manipuler automatiquement ces informations, chacune d'elles doit être représentée dans une forme structurée qui rend accessible l'ensemble des éléments la constituant.

Jusqu'à récemment, les méthodes utilisées dans la collecte d'information à partir de textes consistaient à confier à un être humain l'étude d'un ensemble de documents afin de recueillir et de structurer les données contenant des informations pertinentes en regard du but fixé [Lehnert & al. 1994].

Une telle tâche est un travail long, coûteux et fastidieux qui s'avère rapidement titanesque tant la quantité de textes à traiter se révèle colossale. La quantité d'information augmente très régulièrement et met en échec la capacité humaine à lire, comprendre et synthétiser une telle masse de documents.

Confier à plusieurs personnes la réalisation de cette tâche afin de résoudre ce problème d'adéquation temps/quantité de textes, entraîne le risque d'une augmentation significative du bruitage des informations récoltées, la multiplication du nombre d'analystes accentuant les risques d'erreurs d'interprétation. En effet même avec un haut niveau d'expertise sur les domaines et sur les informations à rechercher, il subsiste toujours chez chaque analyste une part d'interprétation et de subjectivité qui peut entraîner une altération des résultats.

L'évaluation de la tâche d'extraction est également difficile car l'appréciation de la qualité et de la pertinence des informations extraites connaît les mêmes soucis de temps, de coût et de subjectivité que l'exécution de la tâche elle-même.

L'accroissement du nombre de documents électroniques et des capacités de traitement électronique de l'information (augmentation de la taille des mémoires et de la vitesse des systèmes) ont imposé le principe d'automatisation de la tâche d'extraction et ont fait émerger les recherches en Extraction d'Information.

[3] En France, de nombreuses lois sont élaborées après l'étude de plusieurs rapports d'expertise dans lesquels des problèmes sont exposés et des solutions suggérées.

L'Extraction d'Information dispose du potentiel nécessaire pour extraire des informations avec nettement plus de rapidité que la collecte réalisée par des humains. Les travaux de C. A. Will [Will 1993a, 1993b] ont montré que la réalisation de la tâche d'extraction par des processus automatiques produit des résultats dont la qualité, mesurée en terme de précision[4] et de taux d'erreurs, est comparable et même parfois supérieure à celle des résultats de travaux menés par des humains, même s'il s'agit d'analystes entraînés spécifiquement.

La tâche d'extraction est néanmoins une activité qui souffre de nombreuses difficultés liées à la nature même de sa matière première, la langue naturelle : la flexibilité du langage (il existe de très nombreuses manières d'exprimer la même idée), ses ambiguïtés (une même expression peut signifier des notions différentes), son dynamisme, sa dimension diachronique (apparitions de néologisme, migration de termes d'un domaine vers un autre, modification du sens de certains mots à travers le temps [Tartier 2000]) rendent cette tâche délicate [Gaizauskas 2002].

Le domaine de l'Extraction d'Information intègre de plus un grand nombre de sous-problèmes non-triviaux d'analyse de la langue comme la recherche de termes ou l'identification de relations sémantiques ou syntaxiques entre entités.

En réponse aux besoins et aux difficultés évoquées précédemment, de nombreux chercheurs en TALN se sont tournés vers la réalisation d'applications d'Extraction d'Information et ont lancé des projets de recherche autour des exigences particulières de cette activité. Ces recherches ont produit des systèmes, des expérimentations et des méthodes qui situent l'Extraction d'Information comme un processus de traitement automatique des textes utilisable dans des applications pratiques [Cardie 1997].

De telles applications concernent des domaines très variées : les assurances [Glasgow & al. 1997], la médecine (extraction de diagnostics, de symptômes ou de traitements sur un patient afin d'aider les médecins et les personnels soignants [Soderland & al. 1995a,b]), le droit (aide à la classification de documents légaux [Holowczak & Adam 1997]), la finance (analyse d'articles journalistiques pour rechercher les fusions et acquisitions d'entreprises [Sundheim 1993a]), ou encore le support technique informatique (projet Astuxe d'amélioration des services d'assistance technique en ligne par l'extraction d'information à partir de messages de demande d'intervention [Poibeau 2002]).

1.2.2 Un composant de la Fouille de Textes

L'Extraction d'Information s'inscrit comme l'une des activités qui consistent à rechercher, découvrir et traiter des connaissances à partir d'un ensemble de textes. L'ensemble de ces activités constitue le domaine dit de Fouille de Textes[5]. Ce domaine s'inscrit dans le contexte général de l'Extraction de Connaissances[6] (ou EC) qui est, à

[4] Précision : nombre de réponses correctes par rapport au nombre de réponses fournies. Le complément de la précision correspond au bruit.

[5] Text-Mining ou TM

[6] Knowledge Discovery ou KD

l'origine, directement dérivée d'un autre domaine de recherche en informatique, la Fouille de Données[7] [Ichimura & al. 2001].

La Fouille de Données est issue du besoin grandissant d'analyser les larges quantités de données collectées par des entreprises ou des institutions et stockées dans des bases de données. Elle réunit l'ensemble des techniques d'Extraction de Connaissances à partir de bases de données[8] et est définie comme l'élaboration de processus non triviaux d'extraction d'informations valides implicites, nouvelles, et potentiellement utilisables à partir de données structurées [Frawley & al. 1991] [Fayyad & al. 1996].

Les techniques utilisées en Fouille de Données sont particulièrement liées à la façon dont ces informations sont formellement structurées dans les bases de données. Il s'agit par exemple de méthodes inductives ou statistiques afin de construire des arbres de décision ou de régressions non linéaires pour effectuer des classifications [Rajman & Besançon 1997] [Feldman & al. 1998].

La Fouille de Données ne s'exerce que sur des données explicitement et formellement structurées, mais il s'avère que beaucoup d'informations ne sont présentes que dans des textes, sous une forme non structurée. La Fouille de Données ne pouvant répondre au problème de leur extraction, des techniques spécialisées pour collecter et gérer du contenu informationnel à partir des données non structurées contenues dans les textes sont devenues nécessaires. L'ensemble de ces techniques se regroupent sous la dénomination de Fouille de Textes [Rajman & Besançon 1998].

Les principaux buts de la Fouille de Textes sont :

- Le recueil de renseignements fiables par la découverte de termes et de concepts présents dans les textes ;
- Le recueil des relations ou règles d'associations entre les termes et les concepts précédents ;
- La détection de tendances à partir de textes ;
- Le regroupement et la sélection de documents en fonction de concepts ou thèmes communs ;
- La production de résumé (d'un seul document ou d'un ensemble de documents) ;
- La réponse à des questions précises (« *Quand la France a-t-elle gagné la Coupe du Monde de Football ?* ») ou plus générales (« *Parlez-moi de Che Guevara* ») en utilisant une masse de textes comme source d'informations.

Le domaine de la Fouille de Texte réunit et intègre dans ses applications des méthodes d'Extraction d'Information, de Recherche d'Information[9], de Question-Réponse[10], de résumé

[7] Data-Mining

[8] Knowledge Discovery in Databases ou KDD

[9] Information Retrieval ou IR

automatique, de catégorisation de textes, de classification et de routage de documents textuels ainsi que le recours à des agents et des techniques de Fouille de Données.

La capacité de la Fouille de Textes à aller au-delà des analyses classiques de bases de données en se focalisant sur la richesse informationnelle des textes a contribué à une croissance importante de ce domaine. Ce développement est renforcé par l'apport que cette technologie peut offrir à des applications qui prennent aujourd'hui une part essentielle dans la gestion des entreprises modernes comme la Gestion de Relation Client (CRM[11]) [Lefébure & Venturi 2005] et l'Intelligence Economique (Business Intelligence) [Takeda & al. 2001].

L'Extraction d'Information est parfois confondue avec deux autres tâches utilisées en Fouille de Texte : la Recherche d'Information et la tâche de Question/Réponse. Nous décrivons, dans les sections suivantes, ces deux tâches et montrons les liens et différences qui existent entre chacune d'elles et l'Extraction d'Information.

1.2.3 Extraction d'Information et Recherche d'Information

1.2.3.1 La Recherche d'Information : définition

La Recherche d'Information (RI) [Salton & McGill 1983] [Baeza-Yates & Ribeiro-Neto 1999] est issue de la recherche documentaire [Van Rijsbergen 1979] (domaine de la Science de l'Information [Blanquet 1997]) dont le but est de répondre à la question « *comment retrouver, dans un ensemble de documents, ceux qui m'intéressent ?* ». La Recherche d'Information consiste à fournir, à partir d'une large collection de textes, un sous-ensemble pertinent de documents correspondant à une requête donnée par un utilisateur (à l'image d'un moteur de recherche sur Internet). L'utilisateur consulte ensuite ce sous-ensemble de documents afin d'y trouver les informations qu'il recherche. Suivant le système de Recherche d'Information, il peut être assisté dans cette tâche par une classification des documents selon leur pertinence ou par la mise en exergue de termes dans les textes afin de faciliter l'identification des passages intéressants (par exemple par surbrillance, soulignage ou encadrement) [Gaizauskas & Wilks 1998].

Classiquement, un processus de Recherche d'Information se déroule en trois phases [Gaumer 2002][12] (cf. figure 1.1) :

- Modélisation des documents et des requêtes : d'une part les documents de la collection de textes sont modélisés et d'autre part la requête de l'utilisateur est transformée en un modèle en accord avec la représentation choisie pour les documents ;
- Appariement : la modélisation de la requête est appariée avec celle des documents. Le but de cette étape est de déterminer la pertinence d'un document par rapport à la requête afin de sélectionner les documents les plus en adéquation avec celle-ci ;

[10] Question-Answering ou QR

[11] Customer Relationship Management

[12] Voir le chapitre 3 de [Gaumer 2002]

- Production et mise en forme des résultats en fonction de la tâche à effectuer : renvoi de tous les documents ou d'une sélection de documents dans l'ordre décroissant de leur pertinence ; renvoi des documents de manière simple ou accompagnés d'un indice de pertinence ; mise en évidence de l'information via, par exemple, la mise en valeur de certains termes (coloration, soulignement, etc.).

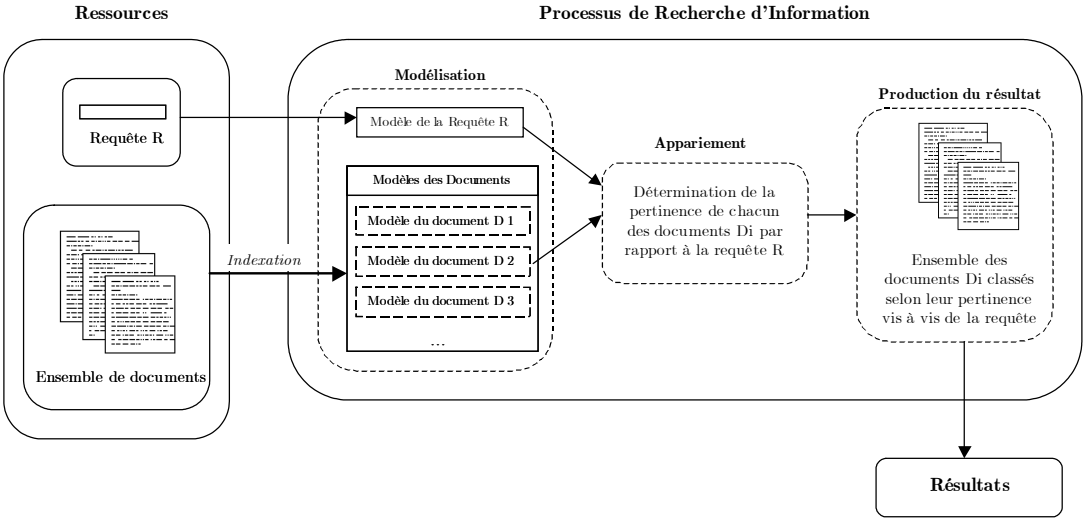

Figure 1.1 : Architecture d'un système de Recherche d'Information

1.2.3.2 Différences et liens avec l'Extraction d'Information

L'Extraction d'Information et la Recherche d'Information poursuivent un but identique (trouver des informations dans un ensemble de textes) mais diffèrent dans leurs réponses et dans les moyens mis en œuvre.

Leur différence fondamentale est la nature de l'information qu'ils renvoient. La Recherche d'Information modélise, de manière indépendante des informations à rechercher, les textes d'une collection de documents, puis sélectionne ceux qui traitent d'un sujet donné (sujet exprimé par une requête), et les fournit à l'utilisateur. Un tel système est ouvert (les requêtes ne sont pas fixées à priori). Les systèmes d'Extraction d'Information effectuent une analyse de documents bruts afin d'en extraire uniquement des informations précises qui intéresseront l'utilisateur, ces informations étant spécifiées à priori [Cunningham 1999] (il n'y a pas de requête en entrée du système).

Par exemple, la réponse fournie par un système de Recherche d'Information à un utilisateur désirant des informations sur les transferts de joueurs concernant le club anglais de football de Manchester United est un ensemble de textes censés contenir les informations qu'il recherche. La lecture de ces documents lui permet d'y trouver les informations qui l'intéressent vraiment (les informations sur les transferts concernant spécifiquement Manchester United). Avec un système d'Extraction d'Information correctement configuré

pour faire une recherche sur le sujet, l'utilisateur obtient des informations précises (par exemple les noms des joueurs transférés, leurs clubs d'origine ou de destination ainsi que les salaires ou indemnités de transfert correspondants).

Ces deux méthodes utilisent des techniques différentes pour des raisons aussi bien pratiques qu'historiques. Les travaux sur les systèmes de Recherche d'Information ont été influencés par la théorie de l'information [Gallager 1968], les théories probabilistes et statistiques [Salton & al. 1975] [Faloutsos & Oard 1996], alors que l'Extraction d'Information est issue de recherches en linguistique computationnelle et en TALN. Les systèmes de Recherche d'Information voient généralement le texte comme un ensemble non structuré de mots. *A contrario*, les systèmes d'Extraction d'Information doivent s'intéresser à la structure grammaticale et aux propriétés syntagmatiques du texte pour éviter d'importantes erreurs de sens.

L'utilisation d'un système d'Extraction d'Information plutôt que de Recherche d'Information pour collecter des informations à partir de textes présente des avantages mais également des inconvénients : d'une part ils sont plus difficiles à mettre en œuvre et sont souvent liés à un domaine de connaissance particulier, ce qui les rend difficilement adaptables à d'autres domaines, et d'autre part les résultats renvoyés sont moins précis que ceux donnés par des lecteurs humains. Mais dans le cas de larges corpus, l'Extraction d'Information apparaît comme potentiellement beaucoup plus efficace que la Recherche d'Information en raison de la difficulté et du coût de la tâche que constitue alors la lecture et l'analyse manuelle de la masse de documents renvoyés par un système de Recherche d'Information, ces systèmes ne se révélant généralement pas assez discriminants. Les systèmes d'Extraction d'Information possèdent également l'atout de pouvoir extraire des faits précis et d'alimenter d'autres applications de traitement de l'information (bases de données, index).

Malgré leurs différences, ces techniques se révèlent complémentaires. L'association de l'Extraction d'Information et de la Recherche d'Information possède en effet un fort potentiel dans la création ou l'amélioration d'applications d'extraction de connaissances à partir de textes. Il existe plusieurs moyens de combiner ces deux systèmes :

- Utiliser la Recherche d'Information en prétraitement de l'Extraction d'Information : face à un très large volume de textes, elle peut fournir à un système d'Extraction d'Information une sous-collection ne regroupant que les documents les plus pertinents. Il existe plusieurs projets dans ce sens comme par exemple le programme TIPSTER [TIPSTER 1993, 1996, 1998]. Ces projets sont encouragés par la masse de plus en plus grande de documents disponibles sur Internet et la difficulté de faire traiter directement de grandes quantités de données textuelles par les systèmes d'Extraction d'Information, les temps et coûts de traitement devenant prohibitifs ;

- Utiliser l'Extraction d'Information pour affiner les résultats d'un système de Recherche d'Information en améliorant la phase de modélisation des documents : les informations extraites de chaque document via un formulaire par un processus d'Extraction d'Information peuvent être utilisées pour créer un index qui modélise le document. Par exemple, le projet Navilex [Pietrosanti 1997] se sert de formulaires d'Extraction d'Information pour indexer des documents légaux ;

- Des techniques propres à l'Extraction d'Information peuvent également être employées afin de compléter les approches classiques de Recherche d'Information pour catégoriser, filtrer et ordonner les documents en fonction de leur pertinence. Un exemple de cette méthode est l'adaptation du système d'Extraction d'Information FASTUS [Hobbs & al. 1996] par John Bear et ses collègues [Bear & al. 1997]. Ce système d'Extraction d'Information attribue une note de pertinence à chacun des documents renvoyés par un système de Recherche d'Information pour un sujet donné, de manière à reclasser ceux-ci en plaçant en tête de liste les documents ayant les meilleures notes. Chaque note est donnée en fonction des informations extraites du texte par FASTUS.

1.2.4 L'Extraction d'Information et la tâche de Question-Réponse

1.2.4.1 La tâche de Question-Réponse

Alors qu'il existe une immense quantité d'informations sous format électronique, il n'y a pas de moyen facile de répondre à une question aussi simple que « *Qui est le sélectionneur de l'équipe de France de football ?* » ou « *Quel pays a connu le meilleur taux de croissance économique en 2001 ?* ». En effet les systèmes d'information les plus usités (les moteurs de recherche sur Internet, les systèmes de Recherche d'Information) ne répondent pas directement à une question précise mais se contentent de fournir à l'utilisateur un ensemble de documents traitant du sujet de la question et contenant potentiellement des éléments de réponse. Le but des systèmes de Question-Réponse (QR) est de résoudre ce problème en cherchant à fournir une réponse précise à une question posée sous la forme d'une séquence linguistique.

Wendy G. Lehnert produit le premier système de Question-Réponse à la fin des années 1970 : le système QUALM [Lehnert 1978]. De petits documents traitant de sujets très spécifiques sont analysés et mémorisés à l'aide d'une représentation conceptuelle. Le système trouve des réponses aux questions posées en consultant cette représentation et en utilisant un raisonnement à partir d'une base de connaissances générales [Ferret & al. 2002a].

Depuis la fin des années 1990, les recherches sur les systèmes de Question-Réponse connaissent un essor important dû principalement à l'avènement du Web [Lin & Katz 2003] et aux conférences TREC[13]. Ces conférences sont organisées tous les ans depuis 1992 par le NIST (*National Institute of Standards and Technology*) et la DARPA (*Defense Advanced Research Project Agency*). L'évaluation de la tâche de Question-Réponse a commencé lors de TREC-8 [TREC 1999] et s'est rapidement imposée comme la tâche la plus populaire de TREC. Les évaluations menées lors des conférences TREC portent sur un ensemble de questions factuelles ou encyclopédiques. La base documentaire utilisée est composée d'environ trois giga-octets de documents électroniques. Les questions et les documents concernent de nombreux domaines (évaluations en domaine ouvert).

Ces conférences, ainsi que le projet AQUAINT de l'armée américaine, ont contribué à la définition de la tâche de Question-Réponse et d'un cadre à la réalisation de systèmes de QR.

[13] Text REtrievial Conference

Ces définitions ont conduit à un accroissement des recherches et au développement de nombreux systèmes, entraînant une amélioration de la qualité des résultats. Ces progrès sont confirmés et renforcés par l'émergence depuis 2000 de nouvelles campagnes d'évaluation : CLEF [14] pour les langues européennes et EQUER [15] [Grau 2002] pour les systèmes traitant le français. Le développement des systèmes de Question-Réponse a également été poussé par les avancées qu'ont connues récemment les technologies de Recherche d'Information [Salton & McGill 1983], d'Extraction d'Information [Piazenza 1997], d'apprentissage [Riloff 1993] ou de reconnaissance des entités nommées [Fourour 2004].

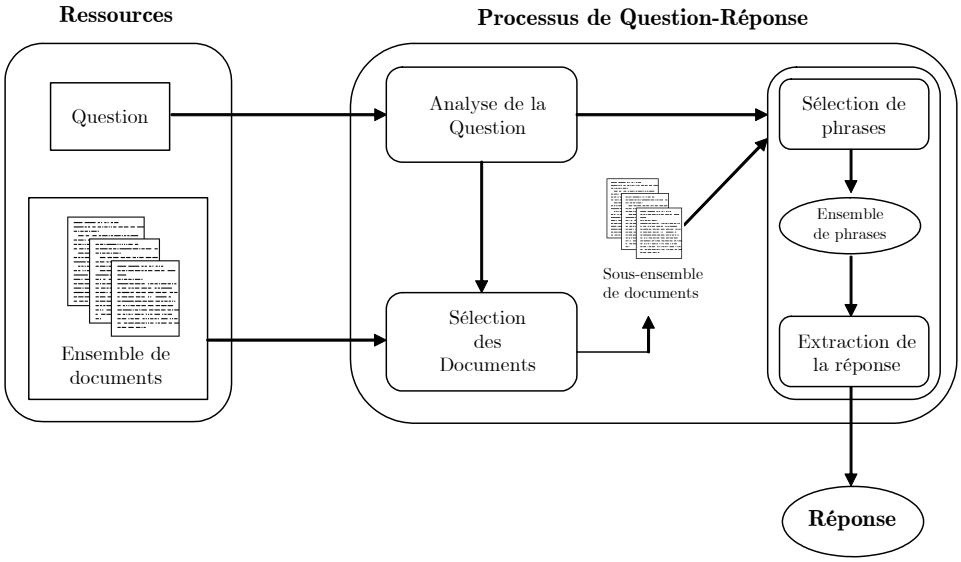

Figure 1.2 : Architecture d'un système de Question-Réponse

Les principaux systèmes de Question-Réponse se décomposent en trois étapes : analyser la question, sélectionner les documents pertinents par rapport à la question et enfin localiser et extraire la réponse dans ces documents (cf. figure 1.2).

1.2.4.1.1 Analyse de la question

En partant d'une question exprimée en langue naturelle, l'analyse de la question [Moldovan & al. 1999] permet d'orienter la stratégie de recherche de la réponse grâce à la détermination des caractéristiques des éléments répondant à la question.

[14] Cross Language Evaluation Forum, `http://clef.iei.pi.cnr.it/`

[15] Évaluation de systèmes de Questions Réponses

L'analyse de la question se déroule classiquement de la façon suivante [Monceaux & Robba 2003] :

- Le processus caractérise le type de la question selon la nature de l'interrogation afin d'identifier le sujet de la question. Des règles sont définies pour typer la question en fonction des éléments introducteurs (pronoms interrogatifs en français, formes en « *wh* » en anglais) ;
- Un processus de reconnaissance des entités nommées détermine ensuite le type attendu de la réponse, c'est-à-dire la ou les classes d'entités nommées auxquelles elle se réfère. Dans ce cas, le processus range ces classes par ordre d'importance. Dans certains systèmes (comme le système QALC [Ferret & al. 1999]) ce processus ne se borne pas aux entités nommées mais cherche également à repérer d'autres classes sémantiques (classes de *Wordnet* [Felbaum 1998] par exemple) ;
- L'étape suivante est la détermination de l'objet de la question (aussi parfois appelé focus), c'est-à-dire de l'élément important de la question [Ferret & al. 2002b]. L'objet exprime ce sur quoi porte la question. Par exemple dans la question « *Quelle est la longueur d'un terrain de football ?* », l'objet est "*terrain de football*". L'objet décrit un concept essentiel de la question [Diekema & al. 2002][Cooper & Rüger 2000] qui est représenté dans les textes par un nom ou un groupe nominal. L'identification de ce groupe nominal est facilitée par la catégorisation du type de la question réalisée précédemment ;
- La dernière étape est l'extraction d'une liste de mots-clefs à partir de la question. Ne sont extraits que les mots considérés comme pertinents, c'est-à-dire ceux qui devraient apparaître dans les documents ou les phrases pertinents vis-à-vis de la question [Alpha & al. 2001].

1.2.4.1.2 Sélection des documents pertinents

Il est nécessaire de restreindre le champ de recherche de la réponse en sélectionnant un sous-ensemble de textes ou de passages de texte pertinents par rapport à la question. Des moteurs de Recherche d'Information sont utilisés dans ce but : soit de façon classique pour collecter un ensemble de documents, soit adaptés de manière à extraire des passages ou des paragraphes de ces documents [Harabagiu & al. 2001]. Les requêtes fournies aux moteurs de Recherche d'Information sont élaborées à partir de la liste des mots-clefs issue de la question. Ces requêtes sont souvent étendues avec des synonymes des mots, généralement en se servant de dictionnaires électroniques. Les documents ou passages sont ensuite classés selon leur pertinence. La plupart des méthodes de classement utilisent une pondération élaborée à partir de statistiques sur le nombre d'occurrences de mots-clés trouvées dans chaque document. Le niveau de correspondance entre les entités nommées attendues et celles présentes dans les textes est également un facteur important dans l'attribution d'un poids à chacun des textes. Les textes dont le poids dépasse un seuil défini par le système sont sélectionnés.

1.2.4.1.3 Localisation de la réponse

La dernière étape consiste à trouver la réponse dans les documents pertinents. Les documents sont découpés en phrases. Ces phrases (phrases-candidates) sont comparées avec la question en se servant des éléments extraits lors de la phase d'analyse de la question. Ensuite une note est attribuée à chaque phrase-candidate. Le calcul de la note fait intervenir une combinaison des méthodes suivantes :

- Les entités nommées présentes dans les phrases candidates sont détectées pour déterminer si leur type correspond avec le type de la réponse attendu ou à une des classes d'entités nommées de la question. Les phrases pertinentes sont celles dans lesquelles cette correspondance est établie. Cette technique est utilisée dans tous les systèmes de Question-Réponse ;

- Les phrases candidates sont appariées avec la question en se servant des termes et de l'objet de la question ainsi que de leurs variantes morphologiques ou syntaxiques voire sémantiques (synonymes, homonymes, hyperonymes, etc.) [Harabagiu & al. 2001]. La qualité des ces appariements est un critère de sélection efficace des phrases-candidates [Attardi & Burrini 2000] ;

- Des patrons d'extraction sont définis à partir de l'objet et de la catégorie de la question [Soubbotin 2001]. Il s'agit généralement de patrons syntaxiques dont la reconnaissance ou non dans la phrase-candidate est caractéristique de la pertinence de celle-ci ;

- D'autres méthodes mettent en jeu des mesures évaluant la correspondance entre les dépendances syntaxiques présentes dans les questions et celles trouvées dans les phrases [Hovy & al. 2001] ou encore des heuristiques sur la structure de la phrase (place des mots, ponctuations) [Cooper & Rüger 2000].

Les phrases les mieux notées sont sélectionnées (phrases-réponses). La réponse est obtenue à partir de ces phrases-réponses (cf. exemple 1.2). Elle prend généralement la forme d'une phrase ou d'un extrait de texte (au nombre de caractères fixé[16]) contenant la réponse ou des éléments de réponse et est assortie d'une valeur de confiance. La réponse peut également être renvoyée sous une forme plus précise où seule l'information demandée (issue du texte) est présente [17] (le nom d'une personne, une date, etc.) [Burger & al. 2001] accompagnée de la référence du document d'où elle est extraite.

Exemple 1.2 (Exemple d'une tâche de Question-Réponse)

À la question : « Qui est le sélectionneur de l'équipe de France de football ? », un système de Question-Réponse sélectionne, à partir d'un ensemble de documents, les phrases-réponses suivantes (phrases ayant obtenu les deux notes les plus élevées) :

Phrase 1 : Raymond Domenech a été nommé vendredi sélectionneur de l'équipe de France par Claude Simonet, président de la FFF.

[16] Lors des conférences TREC, la taille demandée pour les extraits de texte renvoyés comme réponses est passée de 250 à 50 caractères.

[17] Les systèmes récents de Question-Réponse s'orientent de plus en plus vers ce type de résultat.

Phrase 2 : La liste du sélectionneur a été dévoilée ce matin, Raymond Domenech a décidé de jouer la carte de la jeunesse pour ce match amical.

L'extraction de la réponse est réalisée à partir de ces phrases. Le système renvoie soit une de ces deux phrases, soit un morceau de l'une d'entre elles (“Raymond Domenech a été nommé vendredi sélectionneur”) ou seulement l'entité nommée “Raymond Domenech”.

1.2.4.2 Différences et liens avec l'Extraction d'Information

Les systèmes de Question-Réponse visent à obtenir les informations qui répondent précisément à une question. Par cet aspect, la tâche de Question-Réponse est proche de l'Extraction d'Information. Cette proximité est renforcée par le recours à des outils utilisés en Extraction d'Information (reconnaissance d'entités nommées, extraction de terminologie).

Cependant des différences existent entre ces deux domaines dans la façon d'exprimer ce qui est recherché. Dans le cadre de l'Extraction d'Information, la connaissance à extraire n'est pas formulée directement en langue naturelle mais est décrite au préalable par un formulaire dont le système se charge de valuer les champs, alors que pour un système de Question-Réponse elle est exprimée directement à travers une question en langue naturelle. Les résultats obtenus sont également différents : un système d'Extraction d'Information renvoie un ensemble d'informations correspondant aux différents champs du formulaire alors que le résultat d'un système de Question-Réponse doit être une information unique et courte.

Malgré ces différences, le but commun (extraire des informations précises d'une collection de documents textuels) a conduit à améliorer l'efficacité des systèmes de Question-Réponse en utilisant des méthodes d'Extraction d'Information.

Une façon de combiner ces deux domaines est de considérer que les formulaires d'Extraction d'Information représentent une collection de questions prédéfinies auxquelles les informations extraites viennent répondre. Il s'agit alors d'exprimer la question posée au système sous la forme d'un formulaire et de chercher ensuite à remplir ce formulaire par des méthodes mêlant des techniques classiques de Question-Réponse et d'Extraction d'Information [Srihari & Li 2000]. Les informations présentes dans le formulaire sont ensuite réutilisées pour fournir à l'utilisateur la réponse à sa question. Une autre façon de mêler ces deux domaines est d'intégrer des techniques d'Extraction d'Information au cœur du processus de Question-Réponse comme dans le système Q/A d'Oracle [Alpha & al. 2001].

1.2.5 Conclusion

Dans cette section, nous avons effectué une introduction au problème posé par la collecte d'informations à partir de textes en présentant les principaux domaines de recherche s'appliquant à trouver des solutions à ce problème : l'Extraction d'Information, la Recherche d'Information et la tâche de Question-Réponse (cf. figure 1.3). Dans la section suivante, nous nous focalisons uniquement sur l'Extraction d'Information en présentant les évolutions de ce domaine depuis son émergence jusqu'à aujourd'hui.

	Recherche d'Information	***Question-Réponse***	***Extraction d'Information***
Expression de l'information à rechercher	Requête	Question en langue naturelle	Formulaire d'informations
Résultat	Ensemble de documents pertinents	Extrait de texte	Collection de formulaires remplis

Figure 1.3 : Synthèse des tâches d'extraction de connaissances à partir de textes

1.3 De la structuration des textes vers des formulaires d'informations

L'Extraction d'Information trouve ses origines avec l'idée développée dans les années 1950 d'organiser l'information contenue dans un document textuel. Cette idée, ainsi que la volonté affichée par la communauté TALN de développer la compréhension de textes, a guidé tout au long des années 1960 à 1980 des recherches aboutissant aux premiers systèmes d'Extraction d'Information. Le domaine de l'Extraction d'Information a réellement été défini et développé dans les années 1990 sous l'impulsion de la DARPA (*Defense Advanced Research Project Agency*, Etats-Unis) au travers des conférences MUC (*Message Understanding Conference*). Les résultats des évaluations réalisées lors de ces conférences et les protocoles définis par celles-ci pour les travaux en Extraction d'Information ont posé les fondements des recherches menées actuellement dans ce domaine.

1.3.1 Les sources de l'Extraction d'Information

1.3.1.1 Une structuration des textes

L'idée de réduire l'information contenue dans les documents en une structure plus facilement utilisable que le texte brut, est apparue après la Seconde Guerre Mondiale afin de simplifier la quête d'information à partir de textes. Zellig S. Harris [Harris 1951] a avancé la faisabilité de cette formalisation de l'information pour des documents appartenant à des sous-langages comme par exemple les documentations techniques. Cette étude est à l'origine des premiers travaux sur le remplissage automatique de formulaires réalisés à partir du milieu des années 1960 à l'université de New York au sein du « *Linguistic String Project* » [Sager 1981].

Les travaux menés par Naomi Sager jusqu'au milieu des années 1980 avaient pour but d'établir une grammaire computationnelle pour la langue anglaise. L'objectif était d'utiliser cette grammaire pour créer des applications visant à formater les informations de textes écrits en langue naturelle selon une structure s'apparentant à celle d'un formulaire

(*information format*) [Sager 1978]. Ces structures correspondent aux premiers formulaires d'extraction et sont à la base de ceux utilisés aujourd'hui dans les systèmes d'Extraction d'Information. Les structures n'étaient pas définies *a priori* par des experts du domaine concerné par les documents traités mais déduits d'une analyse distributionnelle des textes. Elles étaient remplies par l'étude de documents afin d'alimenter des bases de données dont l'architecture leur correspondait (les champs d'une table correspondant aux champs d'un formulaire). Ces recherches se sont portées sur des textes issus de corpus médicaux et particulièrement sur des documents cliniques [Sager & Lyman 1978].

D'autres projets de structuration de documents textuels ont vu le jour dans les années 1960 pour des tâches telles que la formalisation des entrées d'encyclopédies par un ensemble de formulaires [Grishman 1997].

1.3.1.2 Une volonté de compréhension des textes

Les études sur la compréhension de textes remontent au début des années 1960 [Sabah 2001]. Cette approche cherche à extraire l'ensemble de la connaissance d'un texte, qu'elle soit présente de manière explicite ou implicite. Il s'agit de saisir le sens du texte dans son intégralité et de déceler les nuances de signification ainsi que les objectifs de l'auteur.

Les méthodes de compréhension de texte construisent une représentation logique du texte. Une telle représentation prend la forme d'un ensemble de structures sémantico-conceptuelles obtenues par des analyses syntaxiques et sémantiques du texte et sont représentées sous un formalisme logique ou semi-logique (logique de description, graphes conceptuels [Sowa 1984], etc.). Ces structures peuvent être utilisées par des processus d'inférence afin de calculer des connaissances non explicites dans les textes.

La compréhension de textes peut ainsi être définie comme « *une transduction qui transforme une structure linéaire (le texte) en une représentation logico-conceptuelle intermédiaire, laquelle est ensuite utilisée pour faire des inférences (répondre à des questions, enrichir une base de données, élaborer des résumés ...)* » [Poibeau 2002].

L'idée d'utiliser la compréhension de textes pour collecter l'information présente dans les textes s'est imposée dans les années 1980 comme une théorie intéressante [Poibeau & Nazarenko 1999]. Des recherches dans ce sens ont vu le jour en Question-Réponse avec le système KALIPSOS [Bérard-Dugourd & al. 1988] et en Extraction d'Information avec le système FRUMP [De Jong 1982].

Dans FRUMP, De Jong se fonde sur les recherches en compréhension de textes de Roger Schank [Schank 1975] pour extraire des informations d'articles journalistiques en remplissant des patrons lexico-sémantiques. Quand l'étude du texte ne permet pas d'instancier complètement les patrons, les champs vides sont remplis par des connaissances inférées.

Plus généralement, de nombreux systèmes de compréhension de textes ont été utilisés pour faire de l'Extraction d'Information (comme le système TACITUS [Hobbs & al. 1992]). Mais ces systèmes ont montré leurs limites, liées notamment au caractère trop général de la plupart des systèmes de compréhension qui s'accommode mal de l'étude de domaines spécifiques. Ces limitations ont amené les chercheurs à se consacrer à des systèmes dédiés à la tâche d'Extraction d'Information.

1.3.1.3 Des premiers systèmes dédiés à l'Extraction d'Information

Au cours des années 1980, apparaissent des recherches visant spécifiquement la tâche d'Extraction d'Information, notamment les premiers systèmes commerciaux répondant aux besoins de grandes compagnies. Par exemple, le système ATRANS [Lytinen & Gershman 1986] traite les comptes rendus de transferts inter-bancaires d'argent en utilisant un formalisme issu des travaux de Roger Schank puis de Gerald De Jong ; les systèmes JASPER (agence Reuters) [Andersen & al. 1992] et SCISOR (General Electric) [Jacobs & Rau 1990] travaillent sur des dépêches journalistiques et cherchent à extraire respectivement les gains et dividendes d'entreprises et les fusions et acquisitions de compagnies.

Ces systèmes, qu'ils soient commerciaux ou académiques [Cowie 1983], sont très fortement dépendants de travaux réalisés à la main (pour la définition et le remplissage de formulaires). Ils n'utilisent ni ressources linguistiques majeures comme des lexiques ou corpus externes, ni techniques d'apprentissage ; ils se révèlent coûteux en temps et en ressources humaines, et limités quant à la qualité de leurs résultats.

Le domaine de l'Extraction d'Information n'a acquis sa maturité qu'avec l'émergence des conférences d'évaluation MUC, avec la définition de critères d'évaluation et le développement de nouvelles techniques qui ont permis une amélioration des résultats.

1.3.2 Les conférences MUC

Les conférences MUC ont été initiées par le département de la défense américaine et se sont déroulées à sept reprises entre 1987 et 1998. Elles sont sponsorisées par l'agence américaine ARPA (*Advanced Research Project Agency*) et organisées en Californie par le *Naval Ocean Systems Center*. Ces conférences traduisent la volonté d'un ensemble de groupes de recherche en Extraction d'Information travaillant dans le cadre d'un projet de l'US Navy sur l'étude de messages navals, de définir des méthodes communes et des corpus de référence afin de pouvoir comparer leurs systèmes.

Les conférences MUC prennent la forme d'une compétition dans laquelle les systèmes d'Extraction d'Information confrontent leurs résultats entre eux et avec ceux d'experts. Il s'agit d'évaluer les performances des différents systèmes en leur faisant remplir les mêmes patrons d'informations à partir des mêmes corpus.

Le protocole d'évaluation est le suivant : deux ressources sont fournies aux participants quelques mois avant les conférences (entre un et six mois) : un échantillon de textes (corpus d'entraînement) ainsi que des *scenarii*. Un *scenario* caractérise les éléments à extraire (description de formulaires). Les participants s'appuient sur ces ressources pour adapter leurs systèmes à la tâche à réaliser. Ensuite, chacun reçoit un ensemble de nouveaux textes à traiter. Les formulaires automatiquement remplis par les systèmes sont renvoyés aux organisateurs. Ces résultats sont comparés avec les mêmes formulaires remplis par des experts (formulaires de référence ou *answer-key*). Les évaluations sont réalisées grâce à un ensemble de mesures issues de la Recherche d'Information.

L'exemple 1.3 présente un extrait de corpus et un formulaire utilisés lors d'une campagne d'évaluation MUC (en l'occurrence MUC-3).

Exemple 1.3 (Corpus et Formulaire MUC)

Bogota, 30 Aug 89 (Inravision Television Cadena 2) - Last nigth's terrorist target was the Antioqua liqueur plant. Four powerful rockets were going to explode very close to the tanks where 300,000 gallons of the socalled catille crude, used to operate the boilers, is stored. The watchmen on duty reported that at 20:30 they saw a man and a woman leaving a small suitcase near the fence that surrounds the plant. The watchmen exchanged fire with the terrorists who fled leaving behind the explosive material that also included dynamite and grenade rocket launchers, metropolitan police personnel specializing in explosives, defused the rockets. Some 100 people were working inside the plant.

The damage the rockets would have caused had they been activated cannot be estimated because the caribe soda factory and the Guayabal residential area would have also been affected.

The Antioquia liqueur plant has received threats in the past and maximum security has always been practised in the area. Security was stepped up last night after the incident. The liqueur industry is the largest foreign exchange producer for the department.

Le formulaire suivant est rempli manuellement par des experts à partir du texte ci-dessus et correspond aux informations qui doivent être trouvées par les systèmes participant à l'évaluation MUC-3.

Date of Incident	29 august 1989
Type of Incident	attempted bombing
Category of Incident	terrorist act
Perpetrator: ID of Indiv(s)	"man" "woman"
Perpetrator: ID of Org(s)	-
Perpetrator: Confidence	-
Physical Target: ID(s)	"Antioquia liqueur plant"
Physical Target: Number	1
Physical Target: Type(s)	commercial: "Antioquia liqueur plant"
Human Target: ID(s)	"people"
Human Target: Number	plural
Human Target: Type(s)	civilian: "people"
Target: Foreign Nation(s)	-
Instrument: Type(s)	-
Location of Incident	Colombia: Antioquia (Department)
Effects on Physical Target(s)	no damage: "Antioquia liqueur plant"
Effects on Human Target(s)	no injury or death: "people"

Le signe « - » indique que le champ n'est pas renseigné, c'est-à-dire qu'aucune information n'a été trouvée pour le remplir.

Les premières conférences MUC ont porté sur des corpus composés de messages de la marine américaine et plus particulièrement de rapports d'opérations navales. La première conférence (MUC-1 ou MUCK-1[18], 1987) n'admettait pas de formulaire prédéfini et ne procédait à aucune mesure d'évaluation. Il s'agissait uniquement de tester les systèmes sur des textes communs et d'observer leur comportement sur des messages totalement nouveaux (c'est-à-dire inconnus des chercheurs avant la conférence). Dans la deuxième campagne (MUC-2, 1989) la tâche à accomplir a été fixée : le remplissage de formulaires prédéfinis. Un formulaire d'une dizaine de champs a été défini et les résultats ont été comparés avec des formulaires de référence remplis manuellement. Les premiers critères d'évaluation ont également vu le jour lors de cette conférence.

Lors de la troisième conférence MUC [Sundheim 1991], quinze laboratoires de recherche ont participé à l'évaluation portant sur un corpus de 1300 documents. Ces textes concernaient des récits d'attentats terroristes en Amérique du Sud. Ils étaient formés de dépêches de presse, d'articles journalistiques et de comptes rendus d'informations radiophoniques ou télévisuels, traduits de l'espagnol par l'institution américaine *Foreign Broadcast Information Service*. Chacun des systèmes a fonctionné sur un ensemble de 100 textes, le but étant de remplir un formulaire de 18 champs. Les éléments à extraire sont des informations sur chaque attentat décrit dans un article (lieu, date, type, cible, nombre de victimes, etc.).

Cette campagne d'évaluation est plus ambitieuse que les précédentes tant en termes de complexité des formulaires que de taille du corpus de test. Cette ambition s'est accompagnée du développement de techniques plus fines d'évaluation avec l'introduction de critères formels fondés sur les mesures de Rappel et de Précision utilisées en Recherche d'Information. Ces mesures sont adaptées au domaine de l'Extraction d'Information pour pouvoir apprécier le silence (information non extraite) et le bruit (information extraite mais non pertinente ou erronée) lors d'une tâche d'extraction. Ces deux mesures de rappel et de précision permettent respectivement de juger de la complétude et du niveau de pertinence d'un système ainsi que de sa correction et de son exactitude.

La mesure de Rappel détermine le taux d'informations correctes extraites par rapport au nombre total d'informations pertinentes disponibles dans le texte analysé.

$$Rappel = \frac{N_{correctes}}{N_{total}} \tag{1.1}$$

avec N_{total} le nombre total d'informations attendues (nombre d'information pertinentes dans le texte) et $N_{correctes}$ le nombre d'informations correctes extraites par le système.

[18] La première conférence s'appelait MUCK mais selon les organisateurs, il ne faut chercher aucune signification au K.

La Précision calcule le rapport entre le nombre d'informations extraites qui sont correctes et le nombre total d'informations collectées (correctes et incorrectes).

$$Précision = \frac{N_{correctes}}{N_{correctes} + N_{incorrectes}} \tag{1.2}$$

avec $N_{incorrectes}$ le nombre d'informations incorrectes extraites par le système.

MUC-3 voit disparaître le recours aux systèmes de compréhension de textes pour l'Extraction d'Information (cf. section 1.3.1.2) au profit de systèmes dédiés. Ainsi le système de compréhension TACITUS du SRI[19] sera remplacé lors de la campagne suivante par le système dédié FASTUS [Hobbs & al. 1992] en raison de résultats médiocres, symptomatiques de la limitation des performances des systèmes de compréhension de textes utilisés pour l'Extraction d'Information.

La quatrième conférence MUC [Sundheim 1992a] suit un format très similaire à MUC-3, les modalités de la tâche à accomplir restent les mêmes (ensemble de 100 textes à analyser), le domaine des documents étudiés est identique et les formulaires à remplir demeurent très proches. On observe lors de cette évaluation un plafonnement des performances et une généralisation des méthodes par automates (systèmes FASTUS, CIRCUS [Lehnert & al. 1992]). MUC-4 se caractérise également par l'introduction dans les critères d'évaluation de la notion de F-mesure [Van Rijsbergen 1979]. Cette mesure est une combinaison des mesures de Rappel et de Précision intégrant une variable β dont la valeur permet de pondérer le rapport entre ces deux mesures. Pour une valeur de 1, le Rappel et la Précision ont la même importance. On parlera alors de mesure de « P & R ».

$$F - mesure = \frac{(1 + \beta^2) \times Précision \times Rappel}{\beta^2 \times (Précision + Rappel)} \tag{1.3}$$

avec β la variable de pondération entre Rappel et Précision.

La cinquième campagne MUC [Sundheim 1993a] a été conduite conjointement avec le programme TISPTER[20] [TISPTER 1993]. La tâche d'évaluation se complexifie nettement : la masse des corpus à étudier est beaucoup plus importante et les corpus concernent désormais deux domaines distincts (les créations, rachats et fusions-acquisitions d'entreprises d'une part et secteur micro-électronique d'autre part) dans deux langues (anglais et japonais). Les formulaires sont bien plus complexes que lors des conférences précédentes avec près de 50 champs à remplir et une orientation « objet » : certains champs appelés

[19] Stanford Research Institute

[20] TISPER est un programme de recherche du gouvernement des États-Unis dans les domaines de la Recherche d'Information et l'Extraction d'Information.
`http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/overv.htm`

« références » (*reference*) prennent pour valeur des pointeurs sur d'autre formulaires (jusqu'alors les champs des formulaires n'étaient remplis que par des données textuelles). Les champs des formulaires sont définis selon quatre types : les champs ensemblistes (*set fill*) remplis par la référence à une catégorie choisie parmi un ensemble de catégories possibles, les champs normalisés (*normalised entry*) contenant des données extraites du texte sous un format prédéfini (comme les dates), les champs texte (*string fill*) remplis par un terme issu du texte sans modification (entité nommée par exemple) et les références (*reference*).

Cette cinquième campagne a vu l'éclosion des premières méthodes d'apprentissage par amorçage [Soderland & al. 1995a] mais les principaux systèmes utilisent des techniques d'appariement de patrons [21] (*pattern-matching*) comme dans le système Proteus de l'université de New York [Grishman 1997]. La similitude des méthodes employées entraîne une uniformisation des performances des principaux systèmes.

En raison de la complexité et du nombre d'informations à extraire, la durée d'adaptation des systèmes à la conférence MUC-5 a été fixée à six mois. Les performances des principaux systèmes se situent aux alentours de 55% pour le rappel et 65% pour la précision [Sundheim 1993b]. Pour comparaison, les performances humaines pour la même tâche sont estimées à 75% pour le rappel et à 85% pour la précision [Sundheim 1992b]. Au niveau du temps d'exécution, il faut entre 15 et 60 minutes à un être humain pour remplir un formulaire alors que le temps moyen nécessaire aux systèmes automatiques pour le même travail se situe entre 2 et 4 minutes [Sundheim 1993b].

Malgré ces relatifs bons résultats, se posent des problèmes dus à la difficulté et au coût du travail d'adaptation des systèmes existants. Pour être réellement utilisable, un système d'Extraction d'Information doit pouvoir s'adapter à différents *scenarii*. Or si l'adaptation à un nouveau *scenario* nécessite plusieurs mois d'efforts pour plusieurs concepteurs et experts, les systèmes d'Extraction d'Information resteront très limités et très peu utilisables. Ce problème de la portabilité des systèmes d'un domaine ou d'un corpus à un autre a encouragé l'essor des méthodes par apprentissage et l'émergence de modules réutilisables et indépendants du domaine d'application. Cette notion de module constitue la base des dernières campagnes MUC de 1995 à 1998.

La sixième conférence MUC [Sundheim 1995] marque un tournant dans la tâche d'évaluation. La tâche d'Extraction d'Information est décomposée en quatre sous-tâches correspondant à autant de modules de base pour un système d'extraction ambitionnant d'être portable : la *reconnaissance des entités nommées* (*Named Entity* ou *NE*), la *résolution de coréférences* (*Coreference* ou *CO*), le *remplissage de patrons d'entité* (*Template Element* ou *TE*) et la *description d'événements* (*Scenario Template* ou *ST*). À la différence des précédentes campagnes, les systèmes ne sont plus évalués dans leur ensemble par une analyse des informations extraites mais à travers leur niveau de performance dans une ou plusieurs de ces sous-tâches. Cette évolution vise à permettre l'émergence de techniques nouvelles en ne se focalisant plus seulement sur la tâche difficile qu'est la

[21] L'appariement de patron ou *pattern-matching* est une technique qui compare un patron (patron linguistique, patron d'information, etc.) à un texte pour y rechercher ses occurrences (c'est-à-dire les éléments du texte qui correspondent aux caractéristiques décrites par le patron). Cette technique est principalement utilisée pour remplir les champs variables d'un patron d'information à partir d'un corpus, ou pour découvrir l'identité ou la classe d'un objet inconnu issu d'un texte.

définition d'un système complet mais sur l'élaboration et l'amélioration de sous-processus bien définis. Le corpus de travail est composé de dépêches de presse portant sur les changements de carrière des dirigeants d'entreprises. Afin d'encourager la prise en compte du coût de la portabilité des systèmes, la durée donnée aux participants pour adapter leurs systèmes est limitée à un mois. Cette conférence a vu le développement d'un grand nombre de techniques pour répondre aux problèmes posés par les différentes tâches. Les méthodes proposées sont principalement fondées sur l'apprentissage et l'acquisition automatique ou semi-automatique de ressources.

La dernière conférence MUC s'est tenue en 1998 [MUC-7 1998] et a affiné la segmentation de l'évaluation par le découpage de la tâche de description d'un événement en sous-processus. La recherche de relations entre éléments a été définie comme un sous-processus à part entière (détection de relations ou *Relation Template*). L'Extraction d'Information se décompose alors en cinq tâches[22] : la *reconnaissance des entités nommées* [Chinchor 1997], la *résolution de coréférences* [Hirschman & Chinchor 1997], le *remplissage de patrons d'entité,* la *détection de relations* et la *description d'événements* [Chinchor & Marsh 1998]. Ce dernier découpage est repris dans la définition de nombreux systèmes d'extraction. Cette dernière campagne d'évaluation s'est intéressée à des textes traitant de lancement de satellites et a vu la confirmation de l'essor des techniques d'apprentissage.

Les conférences MUC ont fourni au domaine de l'Extraction d'Information des ressources communes, des outils d'évaluation et surtout une véritable identité en donnant une définition claire de la tâche à effectuer : « *identifier les instances d'une classe particulière d'événements ou de relations dans un texte en langue naturelle, et extraire les arguments pertinents de l'événement ou de la relation* ».

Les systèmes évalués lors des conférences MUC ont nettement occupé le devant de la scène en Extraction d'Information lors des années 1990. Néanmoins plusieurs projets européens d'Extraction d'Information ont été développés avec le soutien de l'Union Européenne lors de cette période. Les projets les plus significatifs sont le système FACILE[23] [Black 1997] et les projets AVENTINUS[24] (application destinée aux services de police de l'Union Européenne) et ECRAN[25] [Cunningham 1996].

[22] Ces différentes tâches sont présentées plus en détail en annexe A.

[23] Fast Accurate Categorisation of Information using Language Engineering

[24] Advanced Information System for Multinational Drug Enforcement

[25] Extraction of content: Research at Near-Market

Chapitre 2

La Note de Communication Orale : un type de textes non-standards

Présentation

Nous effectuons d'abord une présentation générale des textes que nous qualifions de standards et de non-standards (section 2.1), puis nous nous intéressons plus particulièrement à la Note de Communication Orale, un type de textes non-standards sur lequel portent nos recherches (sections 2.2 et 2.3).

2.1 Des textes standards et non-standards

Les textes sur lesquels se focalisent les recherches et les applications de Fouille de Texte ou bien les travaux en Traitement Automatique des Langues Naturelles appartiennent à la langue standard et respectent ses conventions. Il s'agit de textes *standards* : textes issus de la littérature (livres, essais), publications (journaux, magazines, articles scientifiques), textes politiques (discours, programmes électoraux, déclarations gouvernementales) ou documents officiels d'une institution ou d'une entreprise (rapport d'expertise, convention collective, contenu de sites Internet). Ce constat pour la langue française est illustré par exemple par les types de textes traités dans la revue *Lexicométrica* ou dans les articles présentés aux JADT (Journées d'Analyse de Données Textuelles) [Purnelle & al. 2004]. Ces manifestations réunissent des professionnels de la communication et de la fouille de données textuelles ainsi que des chercheurs s'intéressant à la statistique textuelle, à la linguistique de corpus, à l'extraction d'informations à partir de corpus ou encore à l'acquisition de connaissances.

Pourtant de très importantes quantités de textes ne respectent pas les normes de la langue standard. Il s'agit par exemple de petites annonces, de documents formés de notes prises lors d'entretiens ou de réunions, de transcriptions de l'oral, de textes scannés et numérisés (via des OCR[1]) ou encore des textes issus de messageries (forums, messagerie interne à une entreprise ou courriels). Ces textes peuvent comporter beaucoup de déviances linguistiques (fautes orthographiques et/ou grammaticales par exemple).

Ces textes, que nous qualifions de *non-standards* par opposition aux textes répondant aux standards de la langue, se caractérisent souvent par un contenu informatif important en termes de pertinence et de quantité. Pourtant peu de travaux ont pour objet de les traiter, notamment en Extraction d'Information. L'absence d'intérêt pour ces textes et de prise en compte de leurs spécificités dans les recherches en Traitement Automatique des Langues est à l'origine de problèmes d'efficacité de la part des systèmes actuels d'Extraction d'Information lorsqu'ils sont utilisés pour collecter de l'information à partir de telles données textuelles. Nous retrouvons ces observations chez Jean Véronis et Emilie Guimier De Neef dans leurs études sur le traitement automatique des nouvelles formes de communication écrite (NFCE) [Véronis & Guimier De Neef 2004].

2.1.1 Définitions

2.1.1.1 Textes standards

Les textes dits ***standards*** sont définis comme ceux qui respectent les normes d'écriture d'une langue. Il s'agit de textes réguliers dans le sens où ils se caractérisent par une régularité vis-à-vis des normes d'écriture. Ces normes regroupent la norme de la ou des langues dans lesquelles ils sont écrits [Bédard & Maurais 1983] ainsi que les usages modernes.

La norme de la langue correspond à la norme prescriptive, c'est-à-dire l'ensemble des prescriptions fixées et par rapport auxquelles sont évaluables des productions linguistiques. Dans toute culture écrite, la norme prescriptive est fondée sur la langue écrite mais elle subit cependant un processus de renouvellement afin d'intégrer continuellement l'usage moderne. Traditionnellement, on parle de norme de la langue ou de « bon usage ». Cette norme comprend l'ensemble des règles décrites dans les dictionnaires de langue (*Petit Robert* [Rey 2003], *Littré* [Blum & al. 2005], *Grand Larousse* [Guilbert & al. 1985], etc.), les dictionnaires de difficultés (*Multidictionnaire des difficultés de la langue française* [De Villiers 1988]) ou les grammaires (*Bon Usage* de Grevisse [Grevisse 1993], *Bescherelle*). Une norme prescriptive implique la discrimination de ses violations, c'est-à-dire des fautes ou erreurs. Cette discrimination permet de mesurer les écarts présents dans un texte avec la norme et ainsi d'évaluer son degré d'adhésion aux caractéristiques des textes standards [Paquette 1983].

[1] Optical Character Recognition : reconnaissance optique de caractère (conversion d'un fichier image représentant du texte en un fichier texte).

2.1.1.2 Textes non-standards

Les textes dits ***non-standards*** se caractérisent par leurs écarts avec les normes qui régissent les textes standards. Nous déterminons deux ensembles de textes non-standards dont l'intersection est non nulle (un texte non-standard pouvant appartenir aux deux ensembles) : les textes atypiques et les textes dégradés.

Les textes atypiques sont écrits en utilisant des règles spécifiques d'écriture (règles atypiques) différentes de celles qui sont usuellement utilisées dans les textes standards. De telles règles atypiques viennent s'ajouter ou se substituer aux règles usuelles chez les rédacteurs de ce type de textes. Les petites annonces ou les SMS (Short Message Service) sont des exemples de textes atypiques.

Les textes dégradés sont écrits en suivant des règles de rédaction (règles dictant la rédaction des textes standards ou règles atypiques) mais avec de nombreuses transgressions ou altérations de ces règles. Nous parlerons dans ce cas d'une utilisation dégradée des règles de rédaction. Les textes constitués de notes prise lors d'entretiens sont un exemple de textes dégradés : ils sont écrits selon un ensemble de règles (règles de la langue et règles propres à la rédaction de notes) mais sont détériorés en raison de la rapidité imposée par la prise de notes (fautes, simplification de la syntaxe et emploi d'abréviations).

Même s'il est tentant de qualifier de non-standard tout texte présentant des dégradations et de l'exclure ainsi des textes standards, la réalité nous montre que nombre de textes perçus comme standards présentent des écarts face à une norme standard savante (surtout sur Internet) mais il s'agit d'écarts accidentels et non répétés systématiquement. La notion de texte dégradé est liée à la présence d'écarts linguistiques réguliers, répétés et en grande quantité.

Il est envisageable de classer de manière plus fine les textes non-standards en les regroupant par sous-catégories en fonction de leurs homogénéités linguistiques et de leurs conditions de production et d'usage (par exemple les petites annonces, les SMS, etc.). Mais la multitude et la variété des corpus de textes non-standards ainsi que l'émergence régulière de nouveaux types de texte (nouvelles façons de rédiger, langues propres à de nouveaux domaines ou à de nouvelles entreprises) rendent difficile l'obtention d'une telle granularité de classification. Devant ces difficultés, nous n'avons pas cherché à catégoriser finement l'ensemble de tous les textes non-standards. Nous avons restreint ce travail à la sous-catégorie particulière de textes non-standards à laquelle nous nous sommes intéressés dans cette thèse : la Note de Communication Orale (présentée dans la section 2.2).

2.1.2 Particularités des textes non-standards

Dans le cadre de la catégorisation de textes, Douglas Biber et Edward Finegan [Biber & Finegan 1986] ont défini une typologie fondée sur les caractéristiques des textes en les combinant entre elles, créant ainsi une classification multidimensionnelle. Ils différencient le genre de texte (déterminé par des facteurs externes, par exemple le médium, lié au but de l'auteur) du type de texte (déterminé par des facteurs linguistiques internes, lié à la forme du texte). Ces critères sont ensuite utilisés pour identifier les similarités entre textes et les

regrouper selon leur catégorie. Cette distinction genre/type d'un texte correspond à une différenciation entre son contenu et sa forme. Afin de dégager des particularités générales aux textes non-standards, nous intéressons au type du texte, c'est-à-dire sa forme.

D'après les définitions données dans la section 2.1.1, la forme des textes standards est définie comme suivant des règles bien établies alors que les textes non-standards se caractérisent par de nombreux écarts avec ces règles. Ces écarts sont à l'origine de particularités concernant l'exactitude orthographique, le vocabulaire utilisé et la syntaxe du texte. Ces particularités ont été observées lors de l'étude de surface manuelle de plusieurs corpus de textes non-standards (corpus de petites annonces provenant de différents journaux[2], corpus de notes d'entretiens commerciaux et de notes de cours, textes issus de forums et de messagerie interne à une grande banque[3]).

2.1.2.1 Exactitude orthographique

L'exactitude orthographique concerne les phénomènes de *ratés* c'est-à-dire de fautes et d'erreurs sur les mots. Nombre de textes non-standards (les textes dégradés) possèdent une faible exactitude, synonyme d'un grand nombre de ratés. Cette particularité est inhérente au mode de communication, à la provenance des textes (transcription d'oral, textes scannés et numérisés) ou à la rapidité de leur saisie (messages, textes issus de prise de notes).

En reprenant les critères de classification des erreurs présentée par François de Bertrand de Beuvron dans l'introduction de sa thèse [De Bertrand de Beuvron 1992], les phénomènes de ratés se décomposent en trois catégories : les fautes de phonologie (fautes dues à une mauvaise transcription de la suite de sons composant un mot), de morphologie (mauvais emploi des règles de flexion des mots) et de graphie (fautes de frappe dans des textes saisis au clavier qui provoquent des altérations dans la forme des mots).

Ces catégories ne prennent en compte que les écarts orthographiques avec des mots existants, répertoriés dans les dictionnaires ou lexiques : dictionnaires généralistes de la ou des langues utilisées dans le texte[4] ou dans des dictionnaires spécifiques aux domaines abordés par le texte (comme par exemple les dictionnaires médicaux[5] ou juridiques[6]). Aussi les phénomènes de ratés ne concernent pas les mots inconnus comme les acronymes ou les mots très abrégés.

---

[2] *Ouest-France*, *Presse-Océan*, *Libération*.

[3] *Crédit Mutuel*

[4] Comme par exemple, pour le français, le Petit Littré [Beaujean & Littré 2003], le Robert [Rey 2003], le TLFi (*Trésor de la Langue Française informatisé*, `http://atilf.atilf.fr/tlf.htm`) ou le Dictionnaire de l'Académie française (`http://www.academie-francaise.fr/dictionnaire/`).

[5] Par exemple, le *Dictionnaire des termes de médecine (27ème édition)* de Marcel Garnier, Valery Delamare, Jean Delamare et Thérèse Delamare (Maloine, 2002, ISBN 2-224-02737-0) ou le *Dictionnaire des difficultés du français médical* de Serge Quérin (Maloine, 1998).

[6] Parmi lesquels le *Lexique des termes juridiques (14ème edition)* de Raymond Guillien, Jean Vincent, Serge Guinchard et Gabriel Montagnier (Dalloz) ou le *Dictionnaire du vocabulaire juridique* sous la direction de Rémy Cabrillac (Collection Objectif droit, Litec).

2.1.2.2 Vocabulaire

Les textes non-standards présentent quelques particularités concernant leur vocabulaire, c'est-à-dire les mots[7] utilisés dans ces textes : la longueur des mots et le recours à des mots hors-dictionnaire.

2.1.2.2.1 Longueur des mots

Les textes non-standards sont généralement écrits de manière plus simple que les textes standards (textes littéraires par exemple). Ils font peu appel à des mots longs et complexes et utilisent donc plutôt des mots courts ou des mots raccourcis (abréviations).

L'utilisation de mots plutôt courts (c'est-à-dire ayant un faible *coût de production*) s'explique par le besoin d'exprimer le plus d'informations possibles dans un texte de taille limitée et/ou avec un temps de rédaction borné (par exemple dans les petites annonces ou les textes issus de prise de notes).

Cette tendance s'inscrit dans un phénomène général à toutes les langues d'abrègement des mots longs dans le discours : tronquements ("*cinématographe*" devient "*cinéma*" et "*ciné*"), sigles ("*S.N.C.F.*", "*U.R.S.S.*") ou encore certains phénomènes de substitution ("*contremaître*" devient "*singe*", etc.). Ce phénomène s'illustre notamment par les observations conduites par G. K. Zipf à l'Université de Harvard au début des années 1930 qui montrent que l'utilisation des mots par des locuteurs est dirigée par une volonté d'optimiser la relation entre le besoin de diversité et la tendance du locuteur à exercer un effort minimal [Zipf 1965].

2.1.2.2.2 Recours à des mots hors-dictionnaire

Nous appelons mots inconnus des mots dont l'existence n'est pas attesté, c'est-à-dire qui ne sont définis dans aucun dictionnaire ou lexique (dictionnaires généralistes ou spécialisés[8]). Il s'agit de mots inventés et utilisés au sein d'une communauté très réduite de personnes (jargon propre à une entreprise par exemple) ou de mots issus d'une altération volontaire d'un mot connu. Rentrent dans cette dernière catégorie, les abréviations qui sont issues de métaplasmes[9] appliqués à des mots connus. Sont exclus des mots inconnus, ceux qui résultent d'une altération non volontaire d'un autre mot (altération due à des fautes lexicales) ainsi que les acronymes ou entités nommées.

---

[7] Un mot est défini comme une suite de lettres comprise entre deux délimiteurs, transcrivant un son ou groupe de sons d'une langue auquel est associé un sens, et que les usagers de cette langue considèrent comme formant une unité autonome. Le délimiteur est le plus souvent le caractère espace, mais il peut s'agir d'autres éléments comme une suite d'espaces, une tabulation ou un signe de ponctuation (voire même parfois aucun espace comme en chinois ou en japonais).

[8] Une liste de plusieurs dictionnaires et lexiques généraux ou spécialisés en ligne est disponible pour le français à `http://www.usherbrooke.ca/biblio/internet/dictio/dicspfra.htm` et pour l'anglais à `http://www.usherbrooke.ca/biblio/internet/dictio/dicspang.htm` .

[9] Le terme métaplasme regroupe tous les procédés qui altèrent le mot par adjonction, suppression ou inversion de sons ou de lettres.

Les rédacteurs de textes non-standards ont régulièrement recours à des mots inconnus alors que ce phénomène apparaît limité dans les textes standards. Parmi les mots inconnus, ils font particulièrement appel aux abréviations. Ce phénomène est très remarquable dans les textes issus de petites annonces ainsi que dans les textes formés de notes prises lors de réunion ou d'entretien.

Le recours à des mots inconnus s'explique d'une part par le caractère confidentiel de certains types de textes non-standards qui sont rédigés au sein d'une toute petite communauté (un service d'une entreprise ou d'une institution) ayant son propre vocabulaire, et d'autre part par la nécessité dans certains cas d'utiliser des mots abrégés (cf. sections 2.1.2.2.1 et 2.2.2.3).

2.1.2.3 Syntaxe

L'étude de la syntaxe porte sur les phrases du texte. Une phrase est définie comme un syntagme verbal maximal, c'est-à-dire une suite de mots comportant au minimum un verbe et délimitée par des éléments significatifs (signe de ponctuation forte[10], tabulation, nombre important d'espaces). Les particularités évoquées ici concernent la longueur des phrases et la façon dont elles sont écrites (structure interne, ordre des mots).

- Dans un texte standard, les phrases sont souvent longues (supérieures à une vingtaine de mots). Par exemple, dans les journaux, la norme de rédaction d'articles de presse conseille de se limiter à 15-20 mots mais dans la réalité les phrases rencontrées dans la presse dépassent nettement cette longueur [Tremblay 1998]. Il en est de même dans les ouvrages littéraires. *A contrario*, les phrases observées dans les textes non-standards sont souvent courtes, leur taille n'excédant généralement pas une quinzaine de mots. Ce phénomène est particulièrement remarquable dans les transcriptions d'oral, les SMS ou les petites annonces ;

- Les phrases des textes standards sont écrites en respectant les règles de grammaire et de syntaxe des langues utilisées. Les textes non-standards se caractérisent par des écarts significatifs avec ces règles au niveau de la construction des phrases. Il s'agit soit d'une construction de phrase comportant des fautes de syntaxe, soit une construction correcte mais différente des standards de la langue (cas des textes atypiques) qui peut alors être considérée comme incorrecte en regard des normes prescriptives de la langue. Les fautes rencontrées sont les suivantes : ordre incorrect des mots dans la phrase, omissions ou emploi inadéquat d'un ou plusieurs mots. L'emploi inadéquat d'un mot correspond à une utilisation erronée de ce mot, qui n'est alors pas à sa place ou bien superflu. Les fautes concernent principalement les déterminants, les propositions ou les pronoms.

[10] Point, point d'interrogation, point d'exclamation, point-virgule, points de suspension.

2.1.3 Une source de connaissances peu exploitée

L'accès à l'information est un problème essentiel pour la plupart des entreprises et des institutions. Pour y répondre, elles cherchent notamment à exploiter les textes afin d'en extraire des informations intéressantes et pertinentes d'un point de vue commercial, sociologique ou industriel.

Ainsi, ces quinze dernières années, de nombreuses applications d'Extraction d'Information destinées à des acteurs socio-économiques (institutions ou entreprises) ont vu le jour (des exemples de telles applications ont été donnés dans le premier chapitre de cette thèse). Dans la plupart des cas, les informations sont recherchées dans des textes externes aux entreprises ou aux institutions, principalement à partir de documents issus de la presse. *A contrario*, les documents électroniques internes produits par ces acteurs sont, la plupart du temps, mal ou pas du tout exploités alors qu'ils sont susceptibles de véhiculer un grand nombre d'informations [Aussenac-Gilles & Condamines 2001] et que leur quantité s'accroît d'année en année avec la généralisation du *tout électronique* dans la production de textes [Gadet 1997].

Dans les entreprises, il s'agit de rapports internes, de notes techniques ou de service, de notes prises lors de réunions ou d'entretiens, de réponses à des enquêtes (d'opinion, d'habitudes de consommation) ou encore de textes issus de messageries (courriels, courriers internes). De tels textes sont rédigés directement de manière électronique, souvent rapidement (prise de notes, messageries), ou bien sont issus de transcriptions à partir de textes écrits manuellement, transcriptions réalisées soit par un rédacteur humain (opérateur de saisie) soit par un logiciel de reconnaissance de caractères appliqué au résultat de la numérisation du texte concerné via un scanner [Schneider & Renz 2000]. D'autres textes sont automatiquement transcrits au moyen d'outils de traitement du signal transformant une suite de phonèmes en texte. Les entreprises souhaitent extraire les informations de ces documents et les traiter afin de les stocker dans des bases de données sur lesquelles pourront être utilisés des outils de fouille de données, d'extraction de connaissances ou encore d'aide à la décision. Par exemple dans les rapports d'activités ou techniques sont capitalisés l'expérience et le savoir-faire. Leur exploitation peut se révéler essentielle pour le suivi, la maintenance, le développement ou même la sécurité d'installations industrielles. Un cas typique est celui des livres de bord tenus par les agents dans les centrales nucléaires dont l'analyse permet de prévenir les risques d'incident. Dans le domaine commercial, les notes prises lors d'entretiens avec des clients contiennent des renseignements sur leurs habitudes, leurs désirs, leurs attentes ou leurs projets. Ces informations peuvent être stratégiques dans l'optique d'une utilisation commerciale (création de profils client par exemple). Il en est de même pour les enquêtes : en raison de problèmes de coût et d'efficacité, les réponses aux questions ouvertes (champs de questionnaires dits « libres ») ne sont que très rarement dépouillées et analysées alors qu'elles s'avèrent souvent les plus informatives.

Les documents internes ont également une grande valeur pour des institutions (gouvernements, instances internationales, etc.) ou des communautés de domaine (monde hospitalier, judicaire, etc.). Citons par exemple dans le monde hospitalier, plusieurs types de texte (appelées aussi *registres*) appartenant à la catégorisation établie par Pierre

Zweigenbaum dans le cadre de CLEF, un projet de corpus échantillonné pour le français [Zweigenbaum et al. 2001] : les comptes rendus d'hospitalisation (CRH), les comptes rendus opératoires (concernant des interventions chirurgicales) ou de différents examens (imagerie[11], biologie, exploration fonctionnelle[12]), les lettres de sortie (version plus synthétique du compte rendu d'hospitalisation) ou celles visant à adresser un patient à un autre médecin, les prescriptions (ordonnance, demande d'examen) ou les comptes rendus de réunion d'équipe soignante ou d'entretien patient/médecin. De tels textes peuvent être rédigés de manière correcte vis-à-vis de la langue et bénéficier de corrections et de relecture, mais ils sont la plupart du temps issus d'une rédaction rapide employant une syntaxe très épurée et de nombreuses abréviations. Mais ces particularités n'enlèvent rien au contenu informationnel de ces textes dont l'exploitation est d'une grande utilité à la communauté médicale aussi bien pour le suivi d'un patient en particulier (examens, ordonnances, entretiens, etc.) que pour la gestion d'un service (compte rendu de réunion) ou la capitalisation des connaissances d'une spécialité médicale.

Aux types de textes précédents, s'ajoute le cas des petites annonces dont une exploitation efficace présente un intérêt réel aussi bien du point de vue de ceux qui les consultent (afin de trouver précisément ce qu'ils recherchent parmi la masse d'annonces consultables) que de ceux qui les rédigent (afin que leurs annonces soient fournies plus efficacement aux personnes potentiellement intéressées).

Les moyens utilisés pour la rédaction de ces textes (rédaction rapide, transcription manuelle ou automatique de l'oral susceptible d'altérer le texte, taille de texte limitée, utilisation de mots abrégés ou propres à une communauté de rédacteurs, etc.) sont à l'origine d'écarts avec les normes usuelles de la langue et leur donnent ainsi des particularités propres aux textes non-standards (présence de fautes d'orthographe ou d'irrégularités lexicales et/ou syntaxiques, recours à des mots inconnus, etc.).

Ces particularités posent d'importants problèmes aux techniques et outils usuels de Traitement Automatique des Langues qui aboutissent le plus souvent à une dégradation considérable de leurs performances, rendant souvent l'utilisation de ces techniques presque impossible en pratique [Véronis & Guimier De Neef 2004]. La plupart de ces techniques sont en effet étroitement liées aux normes de la langue, parce qu'elles sont observées ou extraites automatiquement à partir de textes rédigés en conformité avec ces normes (corpus journalistiques, scientifiques, littéraires, etc.) et s'accommodent mal de textes s'en détachant.

Aussi, malgré leur caractère fortement informatif, les textes précédents sont souvent laissés de coté car leur exploitation s'avère très difficile. D'une part, leur analyse manuelle est difficile et coûteuse car les volumes à traiter sont souvent très grands et d'autre part, l'analyse automatique ne donne pas de résultats satisfaisants car les méthodes usuelles d'Extraction d'Information ou d'Acquisition de Connaissances, fondées sur des techniques

[11] Comme par exemple la radiologie, les scanners ou les IRM (Imagerie par Résonance Magnétique, une technique de diagnostic fondée sur le phénomène physique de résonance magnétique nucléaire (RMN) qui fournit des images en coupe de grande précision anatomique sans utilisation de rayons X ni d'autres radiations).

[12] Catégorie englobant les examens respiratoires ou neurologiques, les analyses anatomopathologiques, les électrocardiogrammes ou encore les électroencéphalogrammes.

classiques de Traitement Automatique des Langues, s'avèrent peu adaptées et d'une efficacité limitée.

Les constats précédents nous ont incité à nous intéresser au traitement de tels documents. Parmi la variété d'écrits que nous regroupons sous la dénomination de textes non-standards, nous avons axé nos travaux de recherche sur le problème de l'Extraction d'Information à partir d'une catégorie particulière de textes[13] : la Note de Communication Orale. Dans la section suivante, nous définissons cette catégorie de textes, nous présentons ses caractéristiques puis nous détaillons les problèmes qu'elle pose aux techniques usuelles d'Extraction d'Information.

2.2 La Note de Communication Orale

2.2.1 Définition

NOTE, subst. fém. [Trésor de la Langue Française informatisé[14]]

Marque écrite[15].

1. *Phrase courte ou fragmentaire destinée à garder mention de ce qui a été vu, lu ou entendu ou à le reconstituer.*
2. *Texte résumant schématiquement ce qui a été vu, lu ou entendu.*

   → *Prendre quelque chose (un exposé, une conférence, etc.) en note : faire un résumé succinct, retenir les idées principales de quelque chose.*
3. *Commentaire imprimé figurant en marge, en bas de page ou à la fin d'un ouvrage pour faciliter sa compréhension.*
4. *Brève communication écrite.*
5. *Facture, somme à payer.*

Nous faisons référence à l'acception 2 du mot *note* pour définir le terme *Note de Communication Orale*. Nous utilisons ce terme pour désigner la catégorie de textes qui englobe tous ceux correspondant à la définition suivante.

Définition d'une Note de Communication Orale

Une Note de Communication Orale (ou NCO) est un texte rédigé afin de synthétiser les informations véhiculées par des locuteurs dans le cadre d'une communication orale. Par communication orale, nous désignons les entretiens, les conversations (directe, téléphonique, visiophonique), les présentations orales (exposés, cours) ou les réunions. Ce texte est

[13] Le terme *catégorie de texte* se réfère ici à la définition de Douglas Biber [Biber 1989], c'est-à-dire un ensemble de textes possédant des similarités de genre et de type.

[14] `http://atilf.atilf.fr/tlf.htm`

[15] Seules les significations de NOTE comme marque écrite sont citées ici. Ne sont pas abordées celles définissant le mot NOTE du point de vue musical ou comme marque d'appréciation.

constitué d'un ensemble de notes (phrases courtes ou fragmentaires destinées à garder mention de ce qui a été vu, lu ou entendu ou à le reconstituer) prises par un rédacteur lors d'une communication orale. Ces notes doivent permettre de remémorer ultérieurement le contenu informatif de la communication initiale.

Une Note de Communication Orale peut être rédigée par un rédacteur extérieur qui observe la communication sans y prendre part ou par un des acteurs de la communication orale.

La rédaction d'une Note de Communication Orale se caractérise par :

- La *transcription de l'essentiel* : le but d'une Note de Communication Orale n'est pas de rendre compte *stricto sensu* de la totalité de la communication mais de saisir l'essentiel de ce qui est dit. Par exemple lors d'une réunion dans une entreprise ou une institution, il ne s'agit pas de transcrire l'intégralité des débats mais seulement ce qui est important (actions prévues, décisions prises, avis, opinions, etc.). La rédaction d'une Note de Communication Orale nécessite de repérer les éléments importants d'un échange verbal ;

- La *rapidité d'écriture* : une rédaction rapide est essentielle pour permettre au rédacteur de bien suivre le flux d'informations sans se laisser déborder par la rapidité de l'expression orale. En effet la rédaction d'une Note de Communication Orale ne doit pas empêcher le rédacteur d'écouter, éventuellement de participer (dans le cas où le rédacteur est un acteur de la communication) et de comprendre la communication. Cet aspect rend nécessaire la rapidité dans la prise de notes car il est rarement possible au rédacteur d'interrompre la communication pour prendre le temps de finir d'écrire ou pour faire répéter un ou plusieurs protagonistes de la communication ;

- La *limitation de l'écrit* : le volume de l'écrit est volontairement limité dans une Note de Communication Orale. D'une part, la rapidité nécessaire lors de la prise de notes incite les rédacteurs à restreindre le volume de l'écrit afin de gagner du temps. Et d'autre part, la réduction de ce qui est saisi aux informations essentielles entraîne une limitation de la quantité d'écrit.

- L'*absence de relecture immédiate* : la rapidité de l'écriture ne permet pas au rédacteur de vérifier la qualité de ses écrits au moment de leur production. En effet il est impossible de procéder à une relecture partielle en cours de rédaction, procédé largement utilisé lors de l'écriture d'un courriel, d'une lettre ou d'un autre document dactylographié.

La rédaction de Notes de Communication Orale, qui peut être caractérisée par la formule « *rendre compte de l'essentiel avec un maximum de rapidité* », est donc une activité complexe et exige un haut niveau de compétence. Aussi, les auteurs ont recours à des techniques de rédaction spécifiques : omission de mots, simplification de la syntaxe, recours fréquent aux abréviations. Malgré l'utilisation de techniques communes, la formulation d'une Note de Communication Orale reste souvent propre à son auteur. Les moyens utilisés

pour synthétiser rapidement les informations saisies oralement sont semblables sur le fond mais peuvent nettement différer dans leur forme d'un rédacteur à un autre : un auteur utilise généralement ses propres mots et son propre système d'écriture (ses propres abréviations par exemple).

Les différentes techniques utilisées lors de la rédaction de Notes de Communication Orale sont à l'origine de caractéristiques linguistiques propres à ce type de texte. Ces caractéristiques se situent sur quatre niveaux : orthographique, typographique, morphologique, et syntaxique (cf. sections 2.2.2 à 2.2.5).

Avant l'avènement de l'ère numérique, les notes prises afin de transcrire le contenu informatif d'exposés, de cours, de réunions, etc. étaient entièrement écrites à la main et conservées sous la forme de masses de papiers manuscrits. Désormais, de grandes quantités de Notes de Communication Orale produites dans les entreprises ou les institutions, sont stockées dans un format électronique. Ces documents sont rédigés directement de manière électronique, ou écrits à la main puis saisis au clavier ou scannés.

Voici trois exemples de Note de Communication Orale (en français et en anglais).

Exemple 2.1 (Notes de Communication Orale) [16]

A. Exemple en français (1)

> *« Premièrement, Freud essaya des traitements physiques conventionnels comme les bains, les massages, les cures de repos, et d'autres méthodes similaires. Mais après que ceux-ci eurent échoués, il essaya les techniques d'hypnose qu'il avait vu utilisées par Jean-Martin Charcot. Finalement, il s'inspira d'une idée de Jean Breuer en utilisant la communication verbale directe pour obtenir d'un patient non hypnotisé qu'il lui révèle ses pensées inconscientes. »*

Lors de l'écoute de l'extrait précédent d'un exposé sur Freud, un auditeur produira la Note de Communication Orale suivante :

```
Freud 1mnt use phys. trtment, cad bains, massages, etc. Apres echecs
essaye hypnose (de Charcot). Flnt use com verbal direct (de Breuer) pr
obtenir pensees inconsct de patient non hypn.
```

B. Exemple en français (2)

La Note de Communication Orale suivante est issue d'un entretien passée entre un banquier (le rédacteur) et un client :

```
PROPO PRET ACHAT RES SECOND PYRENEE A LUCHON 250KF 180 MOIS
MODULVARIABLE A 5.70 %.
```

[16] Les exemples A et C sont empruntés à un cours sur la prise de notes des *Student Academic Services* de la *California Polytechnic State University* (`http://www.sas.calpoly.edu/asc/ssl/notetaking.systems.html`).

C. Exemple en anglais

```
Mel didn't repr. life as was; e.g., lang. of Ahab, etc. not of real
life.
```

La note précédente transcrit les informations contenues par l'extrait de discours suivant :

> *« Melville did not try to represent life as it really was. The language of Ahab, Starbuck, and Ishmael, for instance, was not that of real life. »*

2.2.2 Caractéristiques orthographiques

La rapidité de l'écriture entraîne l'apparition de très nombreuses fautes d'orthographes. Ce phénomène est aggravé par l'absence de relecture immédiate des écrits par leurs rédacteurs.

Les fautes d'orthographe présentes dans les Notes de Communication Orale se divisent en trois catégories [De Bertrand de Beuvron 1992] : les fautes de phonologie, les fautes de morphologie et les fautes de graphie.

2.2.2.1 Fautes de phonologie

Les fautes de phonologie regroupent les fautes dues à une mauvaise transcription de la suite de sons composant un mot. Ces erreurs phonologiques sont également appelées « fautes d'usage ». Les fautes de phonologie portent souvent sur le doublement de consonnes, des erreurs d'accentuation (accent erroné ou absence d'accent) ou l'emploi d'une syllabe à la place d'une autre dans la transcription d'un phonème.

Ce type de faute est très courant dans les Notes de Communication Orale relatant des discussions orales [Lopez 1999] [Dubreil 2002].

Certaines fautes de phonologie aboutissent au remplacement de mots par des homonymes ou des paronymes. Ces phénomènes peuvent entraîner des problèmes de compréhension du sens des phrases dans lesquelles ils se produisent et même du texte tout entier. On parle dans ce cas d'erreurs cachées (*real-word errors* [Hirst & Budanitsky 2004]), c'est-à-dire de fautes altérant un mot de façon à ce qu'il prenne la forme d'un autre mot existant dans un dictionnaire. Par exemple, dans la phrase "*le livre est* ***sure*** *la table*", "*sure*" est un mot erroné car le mot correct à employer dans cette phrase est "*sur*", il s'agit néanmoins d'un mot qui existe dans la langue. Il faut noter que ces erreurs cachées ne sont pas détectées par les correcteurs orthographiques usuels.

Exemple 2.2 (Fautes de phonologie)

- *Il* ***aporta*** *le journal à son directeur*
  *"a**p**orta" au lieu d' "a**pp**orta" (consonne non doublée)*
- *Ce n'est pas un* ***probléme*** *:*
  *"probl**é**me" pour "probl**è**me" (utilisation du mauvais accent)*
- *Le livreur de* ***laid*** *est passé :*
  *"l**aid**" à la place de "l**ait**" (présence d'un homonyme à la place du mot prévu). Il s'agit ici d'une erreur cachée.*
- *02-05 recu m. julian fait plusieues prposition* ***prèt*** *travaus*
  *"pr**è**t" pour "pr**ê**t" (exemple tiré d'une Note de Communication Orale)*

2.2.2.2 Fautes de morphologie

Les fautes morphologiques résultent du mauvais emploi des règles de flexion des mots, comme les fautes d'accord (genre, nombre) ou de conjugaison.

L'absence de relecture immédiate lors de l'écriture de Notes de Communication Orale favorise la persistance de nombreuses fautes de morphologie qu'une simple relecture permettrait de corriger.

Exemple 2.3 (Fautes de morphologie)

- *erreurs de déclinaison :*
  "travaus" ou *"des canals"*
- *fautes de conjugaison :*
  "ils vient", "on a bue"
- *fautes d'accord :*
  "une femme gêné" (genre) ou "des train" (nombre)

2.2.2.3 Fautes de graphie

Les fautes de graphie correspondent aux accidents dactylographiques, c'est-à-dire aux fautes de frappe présentes dans les textes saisis au clavier.

Elles sont particulièrement fréquentes dans les textes dont la saisie est effectuée rapidement. C'est pourquoi elles constituent la plupart des fautes présentes dans les Notes de Communication Orale rédigées directement de manière électronique.

Ces fautes provoquent des altérations dans la forme du mot. On peut les diviser en plusieurs catégories [Pérennou & al. 1986] :

- Effacement d'une ou plusieurs lettres (D) ;
- Insertion d'une ou plusieurs lettres (I) ;

- Substitution d'une ou plusieurs lettres par d'autres (S). Notons que les fautes de ce type sont généralement dues à la proximité des deux touches sur le clavier (comme le *I* et le *U* sur les claviers AZERTY par exemple) ;
- Permutation de deux lettres consécutives (T) ;
- Coupure d'un mot par un espace (Σ) ;
- Assemblage de deux mots consécutifs (Γ).

Pour un même mot, il est bien sur possible de trouver différents types de fautes.

Les fautes de graphie peuvent être à l'origine d'erreurs cachées et, par conséquent, de problèmes d'ambiguïté au niveau du sens des phrases et du texte. Par exemple "*le sermon du* ***ciré*** " à la place de "*le sermon du* ***curé*** ", le mot "*curé* " devenant "*ciré* " par la substitution de la lettre U par un I (erreur de type S).

Exemple 2.4 (Fautes de graphie)

- ***remplcement*** *de M. Buchet (D)*
  "*remplcement*" *à la place de* "*remplacement*"
- *La diminution des recettes sera* ***comprensé*** *par un gain sur les coûts (I)*
  "*comprensé*" *pour* "*compensé*"
- *Donne le* ***nombxe*** *de sac dans cet ensemble (S)*
  "*nombxe*" *au lieu de* "*nombre*"
- *Il y a 400 personnes dans ce* ***téhâtre*** *(T)*
  "*téhâtre*" au lieu de "*théâtre*"
- *La* ***boulan gerie*** *est ouverte* (Σ)
  "*boulan gerie*" *pour* "*boulangerie*"
- *La reprise de 30kf* ***permettraun*** *remboursement de la différence* (Γ)
  "*permettraun*" au lieu de "*permettra un*"
- *J'ai évoqué la* ***possibilré***
  "*possibilré*" *à la place de* "*possibilité*", *erreurs de type D (suppression du* "*i*" *entre* "*l*" *et* "*é*") *et S (*"*r*" *à la place de* "*t*").
- *02-05 recu m. julian fait* ***plusieues prposition*** *prèt travaus*
  "*plusieues* " *à la place de* "*plusieurs* " *(erreur de type S :* "*e* " *à la place de* "*r* ") *et* "*prposition*" *à la place de* "*proposition*" *(erreur de type D : effacement du premier* "*o*").

2.2.3 Caractéristiques typographiques

La rapidité nécessaire à la rédaction de Notes de Communication Orale est à l'origine de particularités typographiques au niveau de :

- La casse : les auteurs peuvent utiliser la casse différemment de l'usage (c'est-à-dire de l'écriture des mots en minuscule sauf pour le premier mot d'une phrase, les noms propres, les sigles, etc.). Certains auteurs écrivent leurs notes entièrement en minuscules, soit entièrement en majuscules, pour gagner du temps (afin de ne pas perdre de temps à passer sur le clavier de minuscule à majuscule et inversement). La mise en majuscule de mots ou de groupes de mots est également employée afin de mettre en relief les informations les plus importantes ;

- Les diacritiques : dans le but de gagner du temps, les caractères accentués sont souvent négligés et remplacés dans les mots par leur équivalent non accentué ("*e*" pour "*é*", "*è*", "*ê*", etc.). Il en est de même pour les caractères avec une cédille (utilisation de "*c*" à la place de "*ç*") ;

- Les apostrophes et les guillemets : à l'instar des accents et des cédilles, les apostrophes et les guillemets sont fréquemment négligés par les rédacteurs. L'omission des apostrophes peut être à l'origine de fautes de graphie de type Γ (assemblage de deux mots consécutifs, cf. section 2.2.2.3). Par exemple "*dici*" au lieu de "*d'ici*".

2.2.4 Caractéristiques morphologiques

2.2.4.1 Abréviations

La principale particularité morphologique des Notes de Communication Orale est la présence d'un grand nombre d'abréviations, c'est-à-dire de formes raccourcies de mots ou de groupes de mots, utilisées pour leur brièveté.

Une abréviation est formée à partir d'un mot ou d'un groupe de mots en appliquant un ou plusieurs procédés de suppression de lettres ou de sons. L'ensemble de ces procédés est désigné par le terme métaplasme[17]. Ces procédés sont les suivants :

- Aphérèse : suppression d'un phonème ou d'une suite de phonèmes au début d'un mot ("*bus*" pour "*autobus*") ;

- Apocope : suppression d'une ou plusieurs syllabes phonétiques à la fin d'un mot ("*prof*" pour "*professeur*", "*num*" pour "*numéro*") ;

[17] Le terme métaplasme englobe également les phénomènes de métathèse (déplacement de lettres ou de sons dans un mot), de synérèse (prononciation groupant en une seule syllabe deux voyelles contiguës d'un même mot) et de diérèse (dissociation des éléments d'une diphtongue) mais il s'agit de procédés purement phonétiques qui ne rentrent pas en jeu dans la création d'abréviations.

- Élision : suppression de la lettre finale d'un mot ("*ça n' prend pas*" au lieu de "*ça ne prend pas*" ou "*encor*" pour "*encore*") ;
- Syncope : suppression d'une lettre ou d'une syllabe à l'intérieur d'un mot ("*msieurs*" pour "*messieurs*", "*ds*" pour "*dans*").

Une abréviation est souvent générée par la combinaison de plusieurs de ces procédés ou la répétition de l'un d'entre eux. Par exemple, l'abréviation "*cpd*" est obtenue à partir de "*cependant*" par deux syncopes (suppression du "*e*" de "*ce*" et du "*en*" de "*pend*") et une apocope (suppression du "*ant*").

Ces procédés permettent d'abréger un mot ou un groupe de mots en le réduisant à un squelette consonantique ("*rdv*" pour "*rendez-vous*", "*ds*" pour "*dans*", "*lgtps*" pour "*longtemps*", "*cpd*" pour "*cependant*", "*qd*" pour "*quand*", etc.). Ce type d'abréviation est très couramment utilisé car les consonnes possèdent une valeur informative plus forte que les voyelles [Anis 2003]. C'est particulièrement le cas dans les langues où le mot est charpenté autour des consonnes comme le français ou l'anglais (par exemple en anglais "*Limited*" est abrégé par "*Ltd*" et "*Doctor*" par "*Dr*").

La réduction d'un mot à son initiale est également traditionnellement utilisée pour abréger un mot (par exemple "*M*" pour "*Monsieur*", "*p*" pour "*page*"). Mais cette méthode reste généralement limitée « *à quelques unités dans des contextes spécialisés* » [Anis 2003]. Elle est notamment utilisée pour abréger des unités de mesure ou monétaires ("*F*" pour "*Franc*", "*m*" pour "*mètre*" ou "*J*" pour "*joule*", "*A*" pour "*ampère*", etc.), ainsi que des mots-outils[18] comme l'adverbe de négation ("*n veux pas*" pour "*ne veux pas*") ou certains pronoms ("*j'viens*" pour "*je viens*") et déterminants ("*d'textes*" pour "*de textes*"). Notons que le point ou l'apostrophe suffixant généralement ce type d'abréviation ("*M.*", "*j'viens*") sont souvent omis en raison de la vitesse de rédaction.

Enfin, certains mots sont abrégés par une forme lexicale phonétiquement proche du mot mais de taille plus réduite ("*kom*" pour "*comme*").

Les abréviations utilisées dans les Notes de Communication Orale sont des abréviations dites officielles (comme les abréviations reconnues par le Système International des unités[19]), des abréviations communes, c'est-à-dire assez couramment usitées (comme "*Mlle*" pour "*Mademoiselle*" ou "*RDV*" pour "*rendez-vous*"), ou plus généralement, des termes issus d'un système d'abréviations propre au rédacteur ou à un petit groupe de rédacteurs (dans un service d'une entreprise par exemple).

[18] Les mots outils (parfois aussi appelés *mots fonctionnels*) sont définis par opposition aux mots sémantiquement pleins (noms, adjectifs, verbes). Il s'agit de la liste des mots que l'on souhaite négliger lors de comparaisons de chaînes (comme par exemple dans les moteurs de recherche sur Internet) : articles, déterminants, prépositions, pronoms clitiques et autres pronoms, conjonctions de subordination et de coordination, certains adverbes, etc.

[19] Une liste des unités de mesure du Système International d'unités (SI) avec leur abréviation officielle est fournie par le Bureau International des Poids et Mesures (BIPM) : `http://www1.bipm.org/fr/si/`

2.2.4.2 Logogrammes

Un logogramme (ou signe-mot) est un signe écrit représentant un mot complet indépendamment de la langue. Le recours à des logogrammes est un moyen courant pour abréger l'écriture de certains mots ou groupes de mots. Dans les textes électroniques, il s'agit de signes-mots isolés ou de séquences de signes-mots [Anis 2003] formés par un ou plusieurs caractères numériques (" 0", " 1", ... , " 9") ou non-alphanumériques[20] (" &", " +", " >", " €", .etc.). Ils sont utilisés dans les Notes de Communication Orale pour :

- Les opérateurs de calcul ou de comparaison (" *+*" pour "*plus*", " *>*" pour "*supérieur à*", etc.) ;
- Des connecteurs (c'est-à-dire des mots ou groupes de mots dont la fonction est de lier des énoncés) exprimant des relations comme la cause ou la conséquence (par exemple "*x -> y*" pour "*x implique y*") ;
- Certaines unités : lorsqu'un signe existe pour représenter une unité celui-ci est généralement utilisé (par exemple le signe "€" pour "*euro*", " ° " pour "*degré*") ;
- Les nombres : le recours à des logogrammes pour représenter les chiffres (par exemple le signe " *1* " représentant "*un*" en français et "*one*" en anglais) permet l'écriture numérique des nombres. Cette écriture est préférée à celle en toutes lettres par les rédacteurs de Notes de Communication Orale car elle requiert beaucoup moins de temps.

2.2.4.3 Expressions numériques

Une dernière particularité morphologique des Notes de Communication Orale est l'utilisation récurrente d'expressions numériques. Il s'agit d'expressions composées d'un ou plusieurs nombres écrits sous forme numérique, ainsi qu'éventuellement d'autres unités lexicales comme des mots ou des caractères non-alphanumériques.

Les expressions numériques permettent aux auteurs de faciliter et d'abréger la transcription de certaines expressions :

- Les dates : le recours à des expressions numériques pour les dates permet d'en abréger l'écriture (notamment par la réduction du mois à un numéro). Exemples : "*10-5-2002*" ou "*10/05/02*" pour le "*10 mai 2002*", "*12 2004*" pour "*décembre 2004*" ;
- Les expressions mettant en jeu des unités (distances, durées, montants, etc.) : à la place d'expressions en toutes lettres, les rédacteurs utilisent des expressions mêlant des nombres exprimés numériquement avec des unités réduites à des abréviations ou remplacées par un signe ou une séquence de signes ("*3j*" pour "*trois jours*", "*100 kms*" pour "*cent kilomètres*", "*30°C*" pour "*trente degrés celcius*", etc.). La présence de ce type d'expression est une conséquence directe de l'utilisation de logogrammes et/ou d'abréviations évoquée dans les deux sections précédentes ;

[20] C'est-à-dire tout caractère autre qu'une lettre ou qu'un chiffre.

- Les cardinaux ("*1er*" pour "*premier*", "*2ème*" pour "*deuxième*", etc.) ou adverbes dérivés de cardinaux ("*1mnt*" ou "*1èrement*" pour "*premièrement*").

Les formats utilisés pour écrire les expressions numériques sont nombreux et variés. Ils peuvent nettement différer d'un rédacteur (ou groupe de rédacteurs) à un autre. Néanmoins il existe quelques normes communes à la plupart des rédacteurs pour l'écriture d'expressions numériques. C'est par exemple le cas du séparateur utilisé dans les dates (pour séparer les nombres correspondant au jour, au mois et à l'année) qui sont généralement pris parmi un ensemble fixé de caractères (le point, le slash, l'espace, le tiret ou le caractère vide).

2.2.5 Caractéristiques syntaxiques

Les contraintes inhérentes à la rédaction de Notes de Communication Orale et les moyens utilisés par les rédacteurs pour les respecter ont des conséquences sur la syntaxe de ces documents :

- Structures syntaxiques simples : la volonté de rédiger rapidement et aussi brièvement que possible proscrit l'utilisation de structures longues et très élaborées (relatives, apposées, etc.). Les auteurs cherchent à se limiter à des phrases simples (sujet verbe complément) voire même seulement à des groupes verbaux ou nominaux ;

- Problèmes d'agrammaticalité dus à des erreurs de syntaxe : la rapidité de rédaction peut entraîner de la part du rédacteur des fautes grammaticales involontaires comme l'absence ou l'emploi inadéquat de mots ;

- Omission volontaire de mots : afin d'accélérer la prise de note, certains mots sont volontairement oubliés comme par exemple les déterminants ou les mots de liaisons. Dans les notes prises lors d'entretiens ou de conversations, le sujet d'une proposition verbale peut être omis lorsqu'il s'agit d'un locuteur bien identifié. Dans ce cas l'absence de sujet devant un verbe signifie que le sujet de ce verbe est ce locuteur (par exemple dans une Note de Communication Orale relatant un entretien entre X et Y et rédigée par X, la proposition "*ne désire pas continuer*" signifiera "*Y ne désire pas continuer*") ;

- Ponctuation réduite, inexistante ou erronée : comme les diacritiques (cf. section 2.2.3), la ponctuation est souvent négligée. Ce phénomène peut provoquer des ambiguïtés syntaxique et sémantique : les erreurs ou l'absence de ponctuation peuvent rendre difficile la détermination des limites de certaines phrases et la compréhension de leur sens.

2.3 Des systèmes d'extraction inadaptés aux Notes de Communication Orale

Ces dix dernières années, les recherches en Extraction d'Information, notamment dans le cadre des conférences MUC, ont donné lieu à l'élaboration de systèmes de plus en plus perfectionnés et efficaces. La quasi-totalité des systèmes d'Extraction d'Information sont fondés sur des règles d'extraction, exprimées par des patrons lexico-syntaxiques [Huffman 1995] [Morin 1999a,b]. Leur application permet de repérer des faits (structure lexico-syntaxique exprimant une information) mettant en jeu des entités présentes dans le texte et les relations entre elles, et d'en générer de nouveaux par extension ou par inférences sur les faits existants [Piazenza 1997]. À partir de cette base de faits, sont renvoyés comme résultats les faits les plus pertinents en regard du problème posé. Les principales approches pour écrire des règles d'extraction sont l'approche manuelle (approche par ingénierie des connaissances) et l'approche par apprentissage [Appelt & Israel 1999].

2.3.1 Approche manuelle

Les règles d'extraction sont écrites manuellement par un ingénieur connaissant le domaine éventuellement en étroite collaboration avec un ou plusieurs experts. Cette connaissance du domaine influence fortement les performances du système. Les patrons lexico-syntaxiques ainsi créés sont ensuite mis en correspondance avec les éléments présents dans les textes via un ensemble de transducteurs appliqués en cascade sur chaque texte (comme dans les systèmes FASTUS [Hobbs & al. 1996] ou ECRAN [Cunningham 1996]). Les différents transducteurs forment les étapes du processus d'Extraction d'Information et sont fondés sur des techniques de Traitement Automatique des Langues Naturelles : une phase d'analyse lexicale (assignation d'étiquettes grammaticales aux mots grâce à une analyse morphologique et à l'exploitation de dictionnaires ou lexiques) et de reconnaissance d'objets du domaine (entités nommées, structures lexicales particulières comme les dates, les montants monétaires, etc.) puis une analyse syntaxique du texte identifiant dans les phrases un ensemble de structures (groupes nominaux, verbaux, structures tête-complément, structures fondées sur des entités, etc.). Les informations sont finalement extraites par une mise en relation des patrons avec les structures extraites pour ne garder que les faits pertinents.

Cette méthode se heurte au problème du manque d'efficacité des analyseurs lexicaux et syntaxiques quand elle est confrontée à des Notes de Communication Orale. Nous décrivons ci-après ces techniques d'analyse ainsi que les difficultés que leur posent les caractéristiques linguistiques des Notes de Communication Orale.

2.3.1.1 Analyse lexicale

Une analyse lexicale consiste à rechercher les catégories lexicales (Nom, Verbe, Déterminant, Préposition, etc.) des mots du texte et à assigner à chacun d'eux l'étiquette grammaticale correspondant à sa catégorie lexicale selon le contexte dans lequel il apparaît.

Les programmes réalisant un tel étiquetage (appelés étiqueteurs ou catégoriseurs) [21] utilisent principalement des approches stochastiques [Church 1988] [El-Bèze 1993]. Ils procèdent par l'application successive de deux types de règles : des règles lexicales puis des règles contextuelles.

2.3.1.1.1 Règles lexicales

Les règles lexicales visent à reconnaître les mots du texte et à leur assigner une étiquette grammaticale. Elles ont recours directement à des dictionnaires [Church 1988] et/ou utilisent un analyseur morphologique [Cutting & al. 1992]. Dans le premier cas, il est nécessaire de disposer de dictionnaires de formes fléchies de grande taille (chaque forme fléchie étant accompagnée d'une ou plusieurs étiquettes grammaticales). Pour les Notes de Communication Orale, le recours aux dictionnaires traditionnellement utilisés pour des processus d'étiquetage donne des résultats très médiocres en raison des nombreux problèmes dus aux fautes lexicales et à l'utilisation d'abréviations. L'analyse morphologique utilise des règles sur la forme des mots (par exemple, en français, l'analyse du suffixe d'un mot permet d'émettre des hypothèses sur sa catégorie grammaticale). Elle permet de pallier les lacunes des dictionnaires et est mieux adaptée à des textes contenant des mots mal orthographiés. Néanmoins les analyses morphologiques restent liées à des lexiques généraux et ne peuvent résoudre le problème posé par la présence de mots très altérés ou de mots particuliers comme les abréviations.

2.3.1.1.2 Règles contextuelles

Les règles contextuelles permettent de choisir une catégorie grammaticale parmi plusieurs (opération de désambiguïsation). Elles permettent également d'attribuer une catégorie grammaticale à un mot non reconnu par les règles lexicales. Une règle contextuelle affecte une catégorie à un mot en fonction des étiquettes grammaticales des mots de son contexte local. En fonction des types d'analyseurs lexicaux, les règles contextuelles sont écrites et appliquées en utilisant différentes technologies : modèles de Markov cachés[22], modèles N-grammes, calcul de cooccurrences, automates et transducteurs à états finis[23], règles symboliques locales, etc. Les règles contextuelles sont définies manuellement ou générées automatiquement à partir d'un corpus d'apprentissage.

L'utilisation de règles contextuelles pour l'analyse de Notes de Communication Orale pose deux problèmes majeurs :

- L'application des règles lexicales ne permet pas de disposer d'un texte étiqueté de manière fiable car de nombreux mots ne sont pas reconnus. Or il est nécessaire que la plupart des mots du corpus soient étiquetés correctement pour pouvoir appliquer avec efficacité des règles contextuelles. En effet l'application de règles contextuelles à partir d'un texte mal étiqueté augmente la probabilité de mauvaise catégorisation

[21] Un exemple d'étiqueteur, l'étiqueteur de Brill [Brill 1993], est détaillé en annexe B.

[22] Hidden Markov Model (HMM)

[23] Finite State Automata (FSA) et Finite State Transducer (FST)

des mots. Cette difficulté rend les résultats de l'application des règles contextuelles très incertains ;

- Il est très difficile de se procurer des corpus d'apprentissage permettant de générer des règles contextuelles adaptées à ces textes. Il n'existe pas de corpus de Notes de Communication Orale annoté *a priori* et constituer un tel corpus d'apprentissage représente une tâche coûteuse, difficile et fastidieuse qui doit être reproduite pour chaque corpus à traiter tant les différences peuvent être grandes aux niveaux lexical et syntaxique entre des Notes de Communication Orale d'origines différentes. En effet, d'une part les fautes orthographiques et/ou syntaxiques n'ont pas de caractère systématique et peuvent nettement différer d'un texte à un autre, et d'autre part chaque rédacteur a souvent recours à son propre système d'écriture (cf. section 2.2.1).

2.3.1.2 Analyse syntaxique

Les outils d'analyse syntaxique automatique visent à extraire des syntagmes (nominaux, verbaux, adjectivaux) d'un corpus et/ou à repérer les relations de dépendance entre les différentes entités de la phrase [Bourigault & Fabre 2000].

Lors des trois dernières décennies, de nombreuses recherches se sont portées en Traitement Automatique des Langues Naturelles pour élaborer des outils d'analyse automatique, notamment pour la langue française [Abeillé 1991]. Un état de l'art détaillé ainsi qu'une évaluation poussée des méthodes automatiques d'analyse syntaxique figurent dans la thèse de Laura Monceaux [Monceaux 2002]. En nous inspirant de ses travaux, nous classons les analyseurs syntaxiques en deux catégories : les analyseurs généralistes et les analyseurs robustes.

2.3.1.2.1 Analyseurs généralistes

Les analyseurs dits « *généralistes* » sont issus des premiers travaux dans ce domaine. Ils ont pour ambition de reconnaître et traiter l'ensemble des phénomènes d'une langue, et de fournir une analyse complète pour toute phrase. Ces analyseurs se fondent sur des grammaires formelles comme les grammaires génératives de Noam Chomsky [Chomsky 1957], les grammaires transformationnelles ou, pour les systèmes plus récents, les grammaires d'unification [Abeillé 1993] : grammaires HPSG[24] (grammaires syntagmatiques guidées par la tête) [Pollard & Sag 1982] ou grammaires LFG[25] (grammaires lexicales fonctionnelles) [Bresnan & Kaplan 1982].

La plupart des grammaires ont recours à des structures de traits comme mode de représentation. Elles se fondent sur des lexiques exhaustifs des mots de la langue. Ces méthodes ne sont pas adaptées aux Notes de Communication Orale car leur application nécessite une régularité syntaxique des textes étudiés pour donner de bons résultats et produit des échecs lorsqu'elle est réalisée sur des phrases agrammaticales (mal orthographiées ou grammaticalement incorrectes) ou sur des phrases possédant des

[24] Head-driven Phrase Structure Grammar

[25] Lexical Functional Grammar

caractéristiques linguistiques particulières (abréviations, marque d'hésitation, absence de ponctuation) [Monceaux 2002]. D'une part, les structures grammaticales dégradées présentes dans les textes ne peuvent être correctement unifiées à des structures de traits élaborées pour fonctionner sur l'ensemble des phrases générées par des règles grammaticales correctes (répondant aux normes de la langue). D'autre part, la présence de dégradations lexicales (fautes d'orthographe, de frappe, etc.) ou d'abréviations aggrave les problèmes car ces mots non identifiés ne peuvent être étiquetés.

2.3.1.2.2 Analyseurs robustes

Des analyseurs dits « *robustes* » ont été développés depuis le début des années 1990 pour répondre à de nouveaux besoins comme le traitement de textes possédant des irrégularités ou des particularités linguistiques (documents issus d'Internet) ainsi que l'analyse de très larges collections de textes (pour la recherche documentaire par exemple). Ces analyseurs ont pour objectif de fournir une analyse syntaxique, que le texte soit composé de phrases linguistiquement correctes ou non. Les méthodes utilisées ne sont pas fondées sur des connaissances grammaticales figées mais sur une étude statistique ou linguistique du corpus.

Les techniques statistiques (ou probabilistes) s'appuient sur des règles probabilistes. Chaque règle détermine la probabilité qu'une séquence de mots soit identifiable à une structure syntaxique [Rajman 1995]. Ces règles forment une grammaire probabiliste et sont apprises à partir d'un corpus préalablement annoté manuellement [Charniak 1993] [Daille 1994]. Après un étiquetage visant à affecter une catégorie grammaticale à chaque mot du texte en fonction du contexte dans lequel il apparaît (analyse lexicale), l'analyse consiste à construire des arbres d'analyse en appliquant récursivement les règles probabilistes (en déterminant les structures potentiellement associables à chaque séquence de mots) puis à calculer la probabilité de chaque arbre (grâce à une combinaison des probabilités des règles utilisées) pour ne conserver que celui ayant la probabilité la plus élevée.

Confronté aux Notes de Communication Orale, ce processus rencontre deux problèmes : d'une part, les étiqueteurs utilisés pour réaliser l'analyse lexicale présentent des performances médiocres sur ce type de textes, performances qui ont pour effet de dégrader l'application des règles probabilistes, d'autre part, l'absence de corpus d'apprentissage pour les Notes de Communication Orale ne permet pas de déterminer des règles probabilistes pour ce type de textes.

Les méthodes linguistiques sont fondées sur des formalismes grammaticaux décrivant un ensemble de phénomènes linguistiques. Certains analyseurs produisent une segmentation de la phrase en constituants (groupes syntaxiques comme les groupes nominaux ou les groupes prépositionnels) de manière incrémentale. D'autres procèdent à l'extraction de relations de dépendance entre les mots à travers l'utilisation de grammaires de dépendance fondées sur des lexiques ou des règles de contraintes. Enfin, certains combinent les deux méthodes précédentes pour retourner une segmentation de la phrase en groupes de mots, ainsi que les relations de dépendance entre ces groupes. Les stratégies utilisées visent en général à effectuer une segmentation en unités lexicales minimales grâce à un étiquetage grammatical des mots du texte, à extraire des relations syntaxiques simples puis à circonscrire des

informations plus complexes concernant la segmentation et les relations de dépendance syntaxique.

À l'instar des analyseurs syntaxiques généralistes, les analyseurs robustes utilisant des méthodes linguistiques ne s'avèrent pas adaptés aux Notes de Communication Orale en raison de leur recours à des étiqueteurs afin d'identifier les catégories grammaticales des mots.

2.3.2 Approche par apprentissage

L'approche par apprentissage consiste à générer des patrons d'extraction directement à partir d'un corpus. La démarche la plus répandue est l'apprentissage à partir d'un corpus annoté (apprentissage supervisé). Les méthodes par apprentissage non-supervisé n'utilisent pas de corpus annoté mais se fondent sur une base d'exemples ou cherchent à apprendre directement à partir du texte brut.

2.3.2.1 Apprentissage supervisé

La plupart des systèmes utilisant l'apprentissage se fondent sur un entraînement à partir d'un sous-corpus annoté. Celui-ci est constitué d'un sous-ensemble des documents du corpus dans lequel sont marquées les entités à repérer par le système. Ces documents sont sélectionnés et annotés manuellement par une personne ayant une bonne connaissance du domaine. Le système est entraîné sur ce sous-corpus par l'application d'un algorithme d'apprentissage et, fort des règles apprises, peut ensuite être appliqué sur le reste du corpus (systèmes AUTOSLOG [Riloff 1993], CRYSTAL [Soderland & al. 1995a] ou PINOCCHIO [Ciravegna 2001]).

Les corpus d'entraînement peuvent s'obtenir de deux façons : soit un corpus annoté du domaine concerné (corpus pré-annoté) existe et est disponible [Ciravegna 2001], soit il n'en existe pas et le corpus d'entraînement est alors construit à partir des textes concernés par la collecte d'information.

L'absence de corpus de Notes de Communication Orale préalablement annoté et la difficulté d'en constituer pour ce type de textes, rendent impossible le recours à l'apprentissage supervisé. À ces problèmes s'ajoute l'utilisation de techniques d'analyse linguistique, peu efficaces sur ce type de texte, pour rechercher dans les textes bruts les schémas appris.

2.3.2.2 Apprentissage non-supervisé

Les principales méthodes d'apprentissage non-supervisé utilisées dans les systèmes d'Extraction d'Information cherchent à apprendre directement à partir de textes bruts (système AUTOSLOG-TS [Riloff 1996]) ou se fondent sur une base d'exemples (systèmes LIEP [Huffman 1995] ou WHISK [Soderland 1999]).

L'apprentissage à partir de corpus brut utilise un ensemble de méthodes fondées sur des analyses lexicales et surtout syntaxiques afin d'extraire des patrons exprimant les relations

syntaxiques entre les mots du texte ou entre les classes sémantiques des mots. Or, les caractéristiques linguistiques des Notes de Communication Orale, et surtout leurs faibles qualités lexicale et syntaxique, rendent peu efficace leur traitement par les techniques automatiques d'analyse lexicale et syntaxique.

Les systèmes à base d'exemples rencontrent des problèmes similaires car la correspondance entre les exemples et le texte est effectuée à l'aide de techniques linguistiques. L'établissement d'exemples pertinents à partir d'un corpus représente une difficulté supplémentaire car cette tâche nécessite d'importants volumes de données et un bon niveau d'expertise pour évaluer la qualité et la pertinence des exemples (notion de bon exemple [Yangarber & al. 2000]).

D'autres méthodes utilisent des ressources extérieures comme des lexiques ou des thésaurus afin d'établir des patrons d'extraction à partir des mots du texte mais la faible qualité orthographique des Notes de Communication Orale est difficilement compatible avec le recours à de telles techniques.

2.3.3 Conclusion

Les méthodes classiques d'Extraction d'Information ne s'avèrent pas adaptées au traitement et à l'analyse de Notes de Communication Orale. En effet quelle que soit l'approche utilisée, les particularités lexicales et syntaxiques des Notes de Communication Orale diminuent nettement les performances des techniques linguistiques mises en œuvre (comme les analyses lexicale et syntaxique ou l'apprentissage). De plus l'absence de ressources linguistiques spécialisées (lexiques, corpus annotés) ne permet pas d'adapter les outils linguistiques à ce type de texte.

Chapitre 3

Modéliser l'information : une solution pour l'extraire

Présentation

L'idée principale sur laquelle nous nous sommes appuyé pour développer une méthode d'extraction d'Information adaptée aux Notes de Communication Orale est *la modélisation des informations*. Nous abordons d'abord les motivations de ce choix (section 3.1) puis nous décrivons le mode de représentation des connaissances que nous avons utilisé : les ontologies (section 3.2). Nous détaillons ensuite les principaux travaux utilisant une ontologie dans une tâche d'extraction (section 3.3). Enfin, nous donnons un aperçu des principes de notre méthode d'extraction fondée sur une ontologie et adaptée aux Notes de Communication Orale (section 3.4).

3.1 Pourquoi modéliser ?

« *La modélisation du contenu d'un document offre une vue structurée qui permet d'imaginer des outils plus variés pour leur consultation ou leur exploitation* » [Aussenac-Gilles & Condamines 2001]. Dès lors que les règles ou clés de lecture de son formalisme sont connues, un modèle représentant les connaissances d'un texte exprime de manière intelligible, pour un être humain ou un système automatique, les informations contenues dans le texte. La notion de modèle utilisée ici correspond à une représentation des connaissances telles que la définit Daniel Kayser [Kayser 1997] c'est-à-dire une approximation d'un ensemble de connaissances dans un langage dont les symboles ou les primitives sont exploitables par un système automatique. Le passage d'un texte à une représentation structurée de ses connaissances est une opération complexe. Mais une fois obtenu, le modèle peut être exploité de manière autonome, sans utiliser des connaissances exogènes. L'exploitation d'un tel modèle ne requiert pas de ressources linguistiques (telles que des lexiques ou des grammaires

de langue). Aussi traiter le contenu informatif d'un texte en se tournant vers une analyse non plus du texte lui-même mais d'une représentation de ses connaissances permet de s'abstraire d'éventuels problèmes de non-conformité du document avec les normes usuelles des langues.

La représentation des connaissances du texte sous un format exploitable par un système d'Extraction d'Information permettrait donc d'utiliser ce système sur n'importe quel texte indépendamment de ses caractéristiques linguistiques, y compris sur les Notes de Communication Orale. Aussi, modéliser les informations contenues dans les textes et exploiter la modèle obtenu s'est imposé comme le fondement de notre méthode.

Nous avons choisi de recourir à un mode de représentation des connaissances très utilisé dans le domaine de l'Intelligence Artificielle (et particulièrement en Ingénierie des Connaissances[1]) qui s'avère bien adapté à des applications informatiques traitant de l'information (*information systems* [Guarino 1998]) : les ontologies. Ce type particulier de modèle de connaissances [Paquette 2002] est fondé sur des concepts et des relations. Il est présent dans différents domaines de recherche en informatique [Sure 2003] (Traitement Automatique des Langues [Bourigault & Aussenac-Gilles 2003], Recherche d'Information [Abdelali & al. 2003], Extraction et Gestion de Connaissances [Staab & al. 2001], etc.).

3.2 Une ontologie : un modèle de connaissances

Dans cette section, nous effectuons une présentation générale des ontologies. Nous définissons d'abord la notion d'ontologie d'un point de vue philosophique et au niveau informatique. Ensuite nous décrivons les différents constituants d'une ontologie ainsi que les principaux types d'ontologies. Les éléments spécifiés ici seront réutilisés dans les sections et chapitres suivant. Nous terminons cette présentation en évoquant les méthodes de construction d'ontologie, et particulièrement celles permettant d'élaborer une ontologie à partir d'un texte.

3.2.1 Définition des ontologies

3.2.1.1 La notion d'ontologie

Le terme *ontologie*, du grec *ONTOS* (l'être, ce qui est) et *LOGOS* (sciences), est originaire de la philosophie aristotélicienne il y a plus de 2000 ans. Pour Aristote, l'Ontologie[2] désigne « *la partie de la métaphysique qui s'applique à l'être en tant qu'être, indépendamment de ses déterminations particulières* » (*Petit Robert de la Langue Française*

[1] L'Ingénierie des Connaissances (ou IC) est une branche de l'Intelligence Artificielle (ou IA) qui étudie les concepts, méthodes et techniques qui permettent de modéliser et/ou d'acquérir des connaissances [Charlet & al. 2000]. Les objectifs de l'Ingénierie des Connaissances sont de définir une aide à l'utilisateur (méthodes, outils logiciels, organisation du travail) pour modéliser des connaissances (individuelles ou collectives, explicites ou implicites, stabilisées ou évolutives, expertes ou techniques) et rendre ces connaissances accessibles.

[2] En suivant Nicolas Guarino [Guarino & Giaretta 1995], le terme « Ontologie » avec la lettre « O » écrite en majuscule est utilisé pour désigner la notion d'ontologie en philosophie.

[Rey 2003]). L'Ontologie correspond ainsi à la partie de la philosophie qui a pour objet l'étude des propriétés les plus générales de l'être, telles que l'existence, la possibilité, la durée, le devenir. Il s'agit d'étudier les êtres en eux-mêmes et non tels qu'ils nous apparaissent : « *Au sens strict, la métaphysique c'est l'ontologie, c'est-à-dire l'étude de l'être dans ses propriétés générales et dans ce qu'il peut avoir d'absolu ; c'est l'étude de ce que sont les choses en elles-mêmes, dans leur nature intime et profonde, par opposition à la seule considération de leurs apparences ou de leurs attributs séparés* » [Foulquié 1962].

Dans la pensée contemporaine, l'Ontologie est définie comme une étude du sens[3] de l'être, considéré simultanément en tant qu'être général, abstrait, essentiel et en tant qu'être singulier, concret, existentiel. Elle correspond ainsi à une conception du réel englobant tous ses aspects et tous ses niveaux. Cette définition rejoint celle donnée par Mihai Drăgănescu dans *L'Universalité Ontologique de l'Information* [Drăgănescu 1996] pour qui « *l'ontologie est la branche de la philosophie qui a comme objet ce qui existe sous la forme d'une description abstraite, en insistant sur des catégories, principes et traits généraux* ». La philosophie traite d'une ontologie générale se référant à l'existence entière (matérielle, informationnelle, sociale). L'ontologie générale (l'ontologie du monde matériel) est structurale-phénoménologique, elle se réfère à tous les paliers et zones de l'existence, y compris les niveaux mental, psychique et social.

L'ontologie informationnelle est un sous-domaine de l'Ontologie décrivant les différentes formes d'information : phénoménologique [Sartre 1943], structurale ou encore structurale-phénoménologique. Elle se réfère à toutes les structures informationnelles y compris les structures logiques et mathématiques qui fonctionnent soit dans un ordinateur, soit dans l'esprit humain.

Toujours d'après Drăgănescu, il existe également des ontologies partielles, propres à un domaine précis : physique, chimie, etc. Ces ontologies partielles sont aussi appelées *ontologie de domaine* car elles sont relatives aux différents domaines de connaissances.

3.2.1.2 Les ontologies en informatique

La notion d'ontologie a été introduite en informatique il y a une quinzaine d'années dans le domaine de l'Intelligence Artificielle, et plus spécialement en Ingénierie des Connaissances pour répondre aux problèmes de représentation et de manipulation des connaissances au sein des systèmes informatiques.

Différentes définitions de la notion d'ontologie en informatique coexistent dans la littérature [Guarino 1997]. La définition la plus communément admise est celle énoncée par Thomas Gruber en 1993 [Gruber 1993] : il définit une ontologie comme une spécification explicite d'une conceptualisation (« *An ontology is an explicit specification of a conceptualisation. The term is borrowed from philosophy where an Ontology is a systematic account of Existence. For AI systems, what 'exist' is that which can be represented.* »). Une conceptualisation est une structure sémantique intentionnelle encodant les règles implicites

[3] Le mot « *sens* » se réfère ici à une « *idée intelligible à laquelle un objet de pensée peut être rapporté et qui sert à expliquer, à justifier son existence* » (définition du *Petit Robert de la langue française* [Rey 2003]).

qui contraignent la structure d'une partie de la réalité, c'est-à-dire un phénomène formé par un ensemble de connaissances du « monde réel ». Elle correspond ainsi à une vue abstraite et simplifiée d'un phénomène dans laquelle sont identifiés les concepts pertinents du phénomène (« *A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose* » [Gruber 1993]). Dans une spécification explicite, les composants de l'ontologie (les concepts, les contraintes sur leur utilisation, les relations entre concepts ou encore les axiomes) sont explicitement définis. D'après cette définition, construire une ontologie c'est représenter des objets du domaine mais aussi décider de la manière d'être et d'exister de ces objets.

Willem Nico Borst [Borst 1997] affine cette définition en présentant une ontologie comme une spécification explicite et formelle d'une conceptualisation partagée (« *An ontology is an explicit, formal specification of a shared conceptualization* »). Dans une spécification formelle, l'ontologie est interprétable sans ambiguïté par une machine. L'ontologie doit donc être traduite dans un langage formel et opérationnel de représentation des connaissances, ce processus de traduction est appelé *opérationalisation* [Trichet 1998]. Une conceptualisation est partagée lorsque l'ontologie représente une connaissance issue d'un consensus, cette connaissance n'est pas particulière à un individu mais est admise par l'ensemble d'une communauté.

D'autres auteurs définissent une ontologie en la situant en fonction des bases de connaissances[4] des Systèmes à Base de Connaissances (SBC). Dans ce cas une ontologie peut être définie comme une structure hiérarchique d'un ensemble de termes décrivant un domaine et qui peut être utilisée comme squelette d'une base de connaissances (« *An ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a Knowledge Base.* ») [Swartout & al. 1997]. Une ontologie peut aussi fournir les moyens de décrire explicitement la conceptualisation derrière la connaissance représentée dans une base de connaissances (« *An ontology provides the means for describing explicity the conceptualization behind the knowledge represented in a knowledge base.* ») [Bernaras & al. 1996]. Ces deux définitions marquent la distance entre une ontologie et une base de connaissances.

Dans nos travaux nous suivons les définitions de Thomas Gruber et de Willem Nico Borst, pour qui une ontologie correspond à une représentation conceptuelle et formalisée de connaissances. En effet, ces définitions situent une ontologie comme une forme de représentation des connaissances adaptée à notre objectif : modéliser les informations contenues dans des textes en suivant un formalisme particulier, et exploiter cette formalisation par un système automatique d'Extraction d'Information.

[4] Knowledge Base ou KB

3.2.2 Les composants d'une ontologie

Les ontologies permettent de spécifier les connaissances d'un domaine particulier. Cette spécification des connaissances (ou modélisation des connaissances) se fonde sur un ensemble de constituants : les *concepts*, les *relations*, les *axiomes* et les *instances* [Bozsak & al. 2002].

3.2.2.1 Concepts

Les ***concepts*** sont utilisés pour représenter les objets sur lesquels portent les connaissances à spécifier. Un concept décrit une notion[5] et peut être décomposé en quatre parties : une *intension*, une *extension*, une *terminologie* et un *identifiant*.

- L'*intension* correspond à la définition formelle du concept. Elle présente un ensemble de caractéristiques (propriétés [Bouaud & al. 1995] [Guarino & Welty 2000] et attributs [Gómez-Pérez & al. 1996]) qui expriment la sémantique du concept ;
- L'*extension* correspond à l'ensemble des êtres que le concept englobe. Ces êtres possèdent en commun les caractéristiques définies par l'intension. Chacun de ces êtres correspond à une production du concept. Nous appellerons production lexicale du concept une représentation en corpus d'un élément de son extension ;
- La *terminologie* d'un concept correspond aux termes qui lexicalisent la notion décrite par le concept. Usuellement elle est constituée d'un terme vedette (celui qui est le plus employé ou qui doit être employé) et de ses synonymes et variantes. Notons qu'un terme peut être ambigu, c'est-à-dire qu'il lexicalise plusieurs notions, chaque notion correspondant à un sens du terme. Dans ce cas, il convient au concepteur de l'ontologie de statuer sur le sens à retenir pour le terme dans le contexte de l'ontologie (c'est-à-dire en fonction des concepts présents dans celle-ci) ;
- L'*identifiant* (ou libellé formel) permet d'identifier le concept dans le modèle. Il correspond à une unité lexicale qui nomme le concept. L'identifiant est généralement choisi dans la terminologie du concept (il s'agit en général du terme vedette). Il est néanmoins préférable de distinguer l'identifiant du concept et le terme afin d'éviter les confusions entre les représentations conceptuelles et linguistiques d'une notion. Une solution consiste à utiliser une unité lexicale dérivée du terme vedette comme par exemple le terme vedette préfixé par « concept » ou « C_ ».

[5] Une notion correspond à une idée générale et abstraite d'un élément qui implique ses caractères essentiels. La notion est utilisée par la pensée pour structurer la connaissance et la perception du monde.

Exemple 3.1 (Concept décrivant la notion de *voiture*)

Le concept ***C_ VOITURE*** *décrivant la notion de « voiture » est formé des éléments suivants :*

Identifiant : C_ VOITURE.

Terminologie : voiture, auto, bagnole, automobile, caisse.

Intension : véhicule de transport automobile motorisé à quatre roues et conçu pour transporter de une à six personnes.

Extension : { Xantia immatriculée 9658 FG 44, La Corolla Verso de ma mère, La twingo jaune devant nous, la golf tdi d'Éric. }

3.2.2.2 Relations

Une ***relation*** correspond à un lien s'établissant entre des concepts et décrit un type d'interaction entre ces concepts. Dans les ontologies, les relations sont généralement binaires [Gómez-Pérez & al. 2004]. Une relation peut se décomposer en plusieurs éléments : une *intension*, une *extension*, une *terminologie*, un *identifiant* et une *signature*.

- L'*intension* d'une relation exprime la nature de la relation (équivalente au type d'interaction qu'elle décrit). Elle s'exprime par l'ensemble des attributs et des propriétés communes à toutes les réalisations de cette relation ;
- L'*extension* d'une relation correspond à l'ensemble des réalisations de cette relation dans le domaine modélisé ;
- La *terminologie* d'une relation correspond aux termes qui lexicalisent la notion décrite par la relation. Comme pour les concepts un terme peut être ambigu c'est-à-dire lexicaliser plusieurs relations (par exemple "*appeler quelqu'un*" peut signifier lui téléphoner, lui donner un nom ou le héler pour le faire venir). Dans ce cas, c'est au concepteur de déterminer, en fonction de l'ontologie, à quelle relation le terme doit faire référence ;
- L'*identifiant* identifie une relation par un terme traduisant la nature de la relation. Ce terme est généralement issu ou inspiré d'un ou plusieurs termes de la terminologie ;
- La *signature* d'une relation correspond aux concepts qui peuvent être liés par une réalisation de la relation dans le domaine modélisé. Elle s'exprime par un n-uplet de concepts.

Exemple 3.2 (Relation *être propriétaire de*)

La relation ***être propriétaire de*** *est constituée des éléments suivants :*

Identifiant : être propriétaire de.

Terminologie : possède, a, est le propriétaire de, est la propriétaire de, est propriétaire de.

Intension : lien de possession entre une personne physique ou morale et un objet physique.

Extension : { Isabelle possède un grand appartement, Fred a une clio bleue. }

Signature : {C_PERSONNE, C_OBJET_PHYSIQUE}

NB : dans cet exemple, le concept C_PERSONNE englobe les personnes morale et physique, le concept C_OBJET_PHYSIQUE englobe les objets physiques.

Dans une ontologie, les concepts sont organisés en taxinomie. Une taxinomie est une organisation hiérarchique d'éléments dans un système de classification. Au niveau des concepts, on parlera de réseau conceptuel dans lequel les concepts sont structurés hiérarchiquement. La relation utilisée pour structurer la hiérarchie de concepts est la relation de subsomption (également appelée relation « *sorte de* » ou en anglais « *is a* »). Il existe une relation de subsomption entre un concept A et un concept B telle que A subsume B (B est une sorte de A) lorsque B est plus spécifique que A. L'extension d'un concept B est plus réduite que celle d'un concept A qui le subsume alors que son intension est plus riche que celle de A. Par exemple le concept C_LIVRE subsume les concepts C_ROMAN et C_ESSAI.

3.2.2.3 Axiomes et instances

Les ***axiomes*** servent à modéliser des assertions toujours vraies dans le domaine (acceptées sans démonstration) qui se traduisent sous la forme de propriétés sur les concepts et/ou les relations. Les axiomes sont utilisés pour représenter des connaissances élémentaires du domaine qui ne peuvent être formellement définies par les autres constituants d'une ontologie [Sure 2003]. C'est le cas des fonctions qui imposent des contraintes sur le modèle. Par exemple, « mère-biologique-de » est une fonction contraignant que la mère biologique d'une personne soit toujours une femme. Les axiomes permettent ainsi de restreindre l'interprétation des concepts et des relations dans le modèle.

Les ***instances*** regroupent à la fois les instances des concepts et celles des relations. Une instance de concept est une production du concept dans le monde décrit par l'ontologie. L'ensemble des instances d'un concept forme son extension. Une instance de relation est un n-uplet d'instances de concept (n le nombre de concepts de la signature).

3.2.3 Classification des ontologies

Il existe de nombreuses classifications des ontologies selon des critères variés comme le degré de formalisme, le sujet, l'objectif opérationnel ou la granularité [Mizoguchi 1997] [Uschold 1996]. Quatre principaux types d'ontologies [Guarino 1998] se dégagent : les ontologies de haut niveau, de domaine, de tâche et d'application.

- Les ***ontologies de haut niveau*** (*top-level ontology* ou *upper-ontology*) portent sur des concepts de haut niveau qui décrivent des notions très générales. Elles sont destinées à une grande communauté d'utilisateurs et d'applications et peuvent se diviser en deux catégories : les *ontologies de représentation* et les *ontologies génériques*. Les ontologies de représentation décrivent des notions utilisées dans toutes les ontologies pour spécifier les connaissances. La Frame Ontology [Gruber 1993] du projet Ontolingua [6] en est un exemple. Les ontologies génériques (ou ontologies générales) décrivent des concepts très généraux qui sont indépendants d'un domaine ou d'un problème particulier (espace, temps, événement). Elles peuvent être utilisées pour amorcer la construction d'une ontologie ou pour compléter une ontologie existante mais incomplète. L'ontologie SUMO [Niles & Pease 2001] est un exemple d'ontologie générique développée par le groupe de travail SUO WG (Standard Upper Ontology Working Group) [7] ;

- Les ***ontologies de domaine*** décrivent les connaissances d'un domaine particulier (domaine de l'automobile, du monde hospitalier, etc.). Les concepts et les relations d'une ontologie de domaine renvoient à des objets du domaine. Ils peuvent être obtenus en spécialisant des concepts issus d'une ontologie générique. Les relations d'une ontologie de ce type décrivent des liens inter-concepts qui sont présents et valides dans le domaine considéré. Comme exemples d'ontologies de domaine, nous pouvons citer Menelas[8] [Zweigenbaum & al. 1995] ou Galen[9] [Rector & al. 1994] dans le domaine médical, PhysSys[10] [Borst & al. 1997] pour la Physique, ou encore les ontologies TOVE[11] [Fox 1992] [Kim & al. 1999] et Enterprise[12] [Uschold & al. 1998] dans le domaine de la Mémoire d'Entreprise [13] (domaine ayant fait l'objet de plusieurs recherches en Ingénierie des Connaissances [Dieng-Kuntz 2001] qui ont

[6] *Ontolingua*, `http://www.ksl.stanford.edu/software/ontolingua/`

[7] Ce groupe propose un ensemble de travaux sur les ontologies de haut niveau dans le but d'aider à la réalisation de processus de Recherche d'Information, d'Extraction d'Information ou de Traitement Automatique de la Langue. Une description de leurs travaux est disponible à l'adresse `http://suo.ieee.org/`

[8] Cette ontologie est visible à l'adresse `http://www.biomath.jussieu.fr/Menelas/`

[9] *General Architecture for Languages, Encyclopaedias and Nomenclatures in medicine*, `http://www.opengalen.org/`

[10] `http://ksi.cpsc.ucalgary.ca/KAW/KAW96/borst/kaw96doc.html`

[11] *TOronto Virtual Enterprise,* `http://www.eil.utoronto.ca/enterprise-modelling/tove/`

[12] `http://www.aiai.ed.ac.uk/~entprise/enterprise/ontology.html`

[13] Le terme *Mémoire d'Entreprise* (ou *ME*) désigne l'ensemble du savoir et savoir-faire mobilisés par les employés d'une entreprise pour lui permettre d'atteindre ses objectifs (produire des biens ou des services). Le terme plus générique *Mémoire d'Organisation* indique que cette notion de mémoire peut s'appliquer à n'importe quel type d'organisation, qu'il s'agisse d'une entreprise, d'un service ou département au sein de l'entreprise, ou bien, à une échelle plus petite, d'un projet.

abouti au développement de méthodes et d'outils spécifiques de construction, de gestion et d'exploitation d'ontologies, comme par exemple la méthode SAMOVAR [Golebiowska 2002] pour la mémoire d'un projet de conception de véhicule) ;

- Les ***ontologies de tâche*** décrivent des connaissances liées à une tâche ou une activité particulière (vendre, naviguer, etc.). À l'instar des ontologies de domaine, les concepts des ontologies de tâches peuvent être obtenus par spécialisation des concepts d'une ontologie générique. Les ontologies de tâche rendent explicite le rôle joué par chaque concept dans l'activité modélisée. Plusieurs ontologies de tâches ont été élaborées dans le cadre du projet Ontolingua ;

- Les ***ontologies d'application*** sont les ontologies les plus spécifiques. Elles mettent en jeu des concepts spécifiques à un domaine et à une activité particulière. Une ontologie d'application peut être vue comme une double spécialisation d'une ontologie de domaine et d'une ontologie de tâche. Ici, les concepts décrivent souvent des objets du domaine liés à une activité particulière. On peut citer les travaux sur la modélisation de ressources pour des applications d'e-formation[14] [Benayache & al. 2004] [Chaput & al. 2004]. Les ontologies d'application sont en général utilisées pour élaborer des applications concrètes mais ne doivent pas être confondues avec des bases de connaissances.

Dans le cadre de l'utilisation d'une ontologie au centre d'un système d'Extraction d'Information, le recours à une ontologie d'application apparaît le plus approprié : d'une part car elle permet de représenter le domaine et la tâche qui motivent la rédaction des Notes de Communication Orale à traiter et d'autre part car il s'agit d'un type d'ontologie particulièrement bien adapté à la réalisation d'une application informatique concrète. Aussi nous avons choisi de fonder notre méthode d'extraction autour d'une ontologie de ce type.

3.2.4 Construire une ontologie

3.2.4.1 Étapes de construction d'une ontologie

La construction d'une ontologie est un travail réalisé conjointement par un ou plusieurs ingénieurs (ontologues) et des experts du domaine, ainsi qu'éventuellement de futurs utilisateurs de l'ontologie. De nombreuses méthodes de construction d'ontologie existent mais, de manière générale, ce processus peut se décomposer en trois étapes successives : la *conceptualisation*, l'*ontologisation* et l'*opérationnalisation* [Fürst 2004].

- L'étape de *conceptualisation* consiste à identifier les connaissances du domaine à représenter dans l'ontologie. Ces connaissances sont issues de documents textuels (corpus du domaine, lexiques ou dictionnaires du domaine) et/ou d'experts du domaine. Il s'agit ici de choisir parmi l'ensemble des connaissances du domaine présentes dans les corpus ou chez les experts, celles qu'il convient de modéliser. Les connaissances choisies sont ensuite conceptualisées, c'est-à-dire exprimées sous forme

[14] Enseignement à distance par Internet, en anglais *e-Learning*

de concepts, de relations, d'axiomes ou d'instances. Cette expression est réalisée en langue naturelle la plupart du temps ;

- L'étape d'*ontologisation* consiste à structurer et à formaliser le modèle issu de la conceptualisation et exprimé en général en langue naturelle. Le but de cette étape est d'obtenir une ontologie spécifiant formellement les connaissances à représenter [Kassel 2002]. La formalisation est plus ou moins complète en fonction du formalisme choisi. Cette phase doit être réalisée par l'ontologue aidé des experts ;
- L'étape d'*opérationnalisation* a pour objectif de traduire le modèle dans un langage de représentation dit « opérationnel », c'est-à-dire un formalisme utilisable dans un processus automatique. Cette étape est inutile si le formalisme utilisé dans la phase d'ontologisation est opérationnel.

3.2.4.2 Élaborer une ontologie à partir d'un texte

Nos travaux portent sur la modélisation de connaissances contenues dans des textes dans le but d'en extraire certaines. Parmi les diverses méthodologies permettant d'élaborer des ontologies, nous nous sommes donc particulièrement intéressé aux méthodes de construction d'ontologie à partir de textes [Even & Enguehard 2003]. Seule la première des étapes décrites précédemment (conceptualisation) peut être réalisée (en partie seulement) à l'aide d'outils automatiques. Cette étape est effectuée en utilisant généralement le processus décrit pans le paragraphe suivant.

La plupart des méthodes classiques de construction d'ontologie à partir de textes [Bourigault & Aussenac-Gilles 2003] se fondent sur le contenu des textes pour construire une ontologie en extrayant les concepts ainsi que leurs relations, principalement à partir d'une analyse du texte, même s'il est reconnu que les textes peuvent ne pas constituer l'unique source de connaissances [Aussenac-Gilles & al. 2000a,b]. D'une manière générale, l'élaboration d'une ontologie à partir d'un texte se fonde d'abord sur l'extraction de termes à partir desquels sont définis les concepts de base (primitives conceptuelles [Nobécourt 2000]), puis sur l'identification des relations lexicales entre ces termes dans le texte afin de faire émerger les premières relations inter-concepts [Lame 2000]. La phase suivante consiste à extraire de nouvelles relations entre les concepts ainsi que de nouveaux concepts en s'appuyant sur l'analyse des relations sémantiques entre les termes. Ces étapes aboutissent à la création d'un réseau sémantique de concepts qui doit être validé par un expert du domaine [Bouaud & al. 1995].

Les différentes étapes sont réalisées au moyen d'outils de Traitement Automatique des Langues et particulièrement d'analyses lexicale et syntaxique (outils essentiels dans la détermination de relations lexicales et sémantiques). Mais les spécificités linguistiques des Notes de Communication Orale, auxquelles ces outils sont peu adaptés (cf. section 2.2.3.1), ne permettent pas d'avoir recours à ces méthodes. La phase de définition de concepts de base à partir des termes du corpus reste cependant envisageable à condition que la technique d'extraction terminologique employée ne s'appuie pas sur des méthodes linguistiques incompatibles avec les Notes de Communication Orale.

3.3 Extraction d'Information fondée sur une ontologie

L'idée d'utiliser une ontologie pour extraire des informations à partir de textes a fait l'objet de plusieurs recherches et expérimentations dans les domaines liés au traitement de l'information [Sure 2003] et notamment pour l'élaboration de processus d'Extraction d'Information. Nous décrivons dans cette section les principaux systèmes développés dans ce sens[15], que ce soit pour répondre au problème du remplissage de formulaires à partir de textes bruts, à celui de l'extraction de contenu à partir d'Internet, ou encore pour intégrer ou utiliser un module d'Extraction d'Information comme source dans des outils de gestion de connaissances ou de données multimédia. Après la présentation de l'ensemble de ces systèmes (section 3.3.1), nous procédons à une confrontation de leurs méthodes avec les Notes de Communication Orale (section 3.3.2).

3.3.1 Principaux systèmes d'extraction fondés sur une ontologie

3.3.1.1 Le système LaSIE

Les premières véritables expérimentations de système d'Extraction d'Information fondés sur une ontologie remontent à une dizaine d'années environ avec le développement du système LaSIE (Large Scale Information Extraction) par l'équipe Natural Language Processing du département informatique de l'Université de Sheffield [Gaizauskas & al. 1995]. LaSIE a été élaboré comme un système d'Extraction d'Information générique conçu à l'origine pour réaliser les tâches spécifiées lors de MUC-6 (cf. section 1.3.2). L'architecture de ce système a ensuite été réutilisée sous le nom de VIE (Vanilla Information Extraction) au sein du projet GATE (General Architecture for Text Engineering), un projet de création d'une plate-forme logicielle pour le développement d'outils de Traitement Automatique des Langues Naturelles [Cunningham & al. 2003].

Le fondement de leur approche est la construction d'une représentation du texte ensuite utilisée pour répondre aux différentes tâches MUC (cf. annexe A) [Gaizauskas & Wilks 1998]. Le modèle obtenu, appelé « *modèle du discours* », représente les connaissances contenues dans le texte. La construction du modèle du discours se déroule en trois étapes successives : d'abord un prétraitement lexical, ensuite une étape d'analyse grammaticale et d'interprétation sémantique et enfin une phase d'interprétation du discours. Une dernière phase se sert du modèle du discours pour produire les résultats désirés.

Le traitement lexical commence par une *tokenisation* du texte. Il s'agit d'identifier dans le texte quelles séquences de caractères doivent être traitées comme des tokens. Cette opération concerne notamment la séparation des signes de ponctuation des autres caractères (par exemple dans "*Inc.,*" seront identifiés les tokens "*Inc*", "." et ","). S'ensuit une

[15] Des systèmes d'Extraction d'Information ont parfois recours à des ontologies pour étendre des patrons d'information déjà existants [Poibeau & Dutoit 2002] ou pour améliorer leurs résultats et corriger certains problèmes notamment dans les processus ayant trait aux classes sémantiques (attribution, extension ou manipulation de classes sémantiques) comme dans les systèmes RAPIER [Califf & Mooney 1999] ou CRYSTAL [Soderland & al. 1995a]. Ces systèmes se servent des ontologies comme complément de méthodes classiques d'extraction et non comme l'un des éléments principaux des processus extrayant l'information.

segmentation du texte en phrases puis un étiquetage grammatical utilisant une version modifiée de l'étiqueteur de Brill [Brill 1993] (cf. annexe B) avec un ensemble d'étiquettes spécifiées par les chercheurs de l'Université de Pennsylvanie [Marcus & al. 1993]. Cette première phase comprend également un module d'analyse morphologique qui détermine la racine des noms et des verbes ainsi qu'un processus de reconnaissance des entités nommées [Wakao & al. 1996].

La deuxième phase construit une représentation sémantique de chacune des phrases en un ensemble de prédicats. Cette représentation logique est réalisée par une triple analyse syntaxique de la phrase. La première utilise une grammaire fondée sur les entités nommées. L'interprétation sémantique est menée en parallèle de l'analyse et produit, pour chaque instance de ces entités, une paire de prédicats : un prédicat unaire spécifiant le type de l'entité et un prédicat à deux arguments spécifiant le nom de l'instance (par exemple *person(e10)* et *name(e10, 'Thierry Henry')* pour l'entité *'Thierry Henry'* décrivant une personne). Cette analyse est suivie d'une autre analyse fondée sur une grammaire plus générale reconnaissant les entités grammaticales dans les phrases (groupes nominaux, verbaux ou adjectivaux, groupes prépositionnels, propositions relatives, etc.). Ici aussi l'interprétation sémantique a lieu au cours de l'analyse et crée une représentation logique des relations syntaxiques qui sont identifiées. Une troisième analyse vise à résoudre les problèmes d'ambiguïté que peuvent poser les deux phases précédentes. Elle consiste à étudier les choix offerts par la grammaire, à sélectionner ceux qui couvrent correctement le plus de cas dans le texte, et à ré-analyser les phrases en tenant compte de ces sélections.

Dans la troisième étape, il s'agit d'intégrer les représentations logiques de chacune des phrases dans un réseau sémantique unique et hiérarchique, dit « *modèle du discours* », qui rend compte du discours contenu dans le texte. Ce modèle du discours est construit en associant les informations présentes dans les prédicats issus des phases précédentes avec une base de connaissances appelée « *modèle du monde* » (*world model*). Cette base est constituée d'une ontologie à laquelle est associé un ensemble de structures attribut-valeur. L'ontologie est une hiérarchie des concepts liés à la nature des formulaires d'extraction à remplir par le système. Elle est construite en utilisant XI, un langage de représentation de connaissances fondé sur des graphes conceptuels à héritage multiple [Gaizauskas 1995]. Les concepts décrits dans les formulaires doivent être présents dans l'ontologie afin que le modèle du monde couvre l'ensemble des entités à rechercher dans le processus d'Extraction d'Information. Le modèle du monde peut être vu comme un squelette sur lequel vient se greffer la représentation sémantique du texte étudié afin de produire un modèle particulier au texte. Le modèle du discours est ensuite étendu par inférences. L'extension consiste à dériver de nouvelles classes sémantiques à partir des classes existantes, à résoudre des coréférences entre instances de l'ontologie et à déduire des informations en utilisant les principes de présupposition et de conséquence [Lecomte & Naït-Abdallah 2003].

La dernière phase est constituée d'un module de génération des résultats qui utilise le modèle du discours pour remplir les différents champs des formulaires d'extraction pris en compte dans la définition de l'ontologie et qui renvoie les résultats dans le format requis par l'utilisateur.

3.3.1.2 Le système SynDiKATe

À la fin des années 1990 est élaboré SynDiKATe (*SYNthesis of Distributed Knowledge Acquired from Texts*), un système de compréhension de la langue naturelle permettant d'acquérir des connaissances à partir de textes et de les représenter de manière formelle. Ils proposent d'utiliser les bases de connaissances textuelles construites grâce à ce système comme source pour extraire de l'information à partir de textes [Hahn & Romacker 2000]. Des expérimentations ont été menées à cette fin sur des textes du domaine médical (MEDSynDiKATe) et informatique (ITSynDiKATe).

Le but de leur approche est non seulement d'acquérir les connaissances du texte en extrayant les concepts et les relations qui sont traditionnellement recherchés par les systèmes d'Extraction d'Information classiques mais également d'étendre dynamiquement l'ensemble des patrons de connaissances par un apprentissage incrémental de concepts. D'après les auteurs, l'intégration de mécanismes d'apprentissage dans le processus de création de la base de connaissances entraîne une augmentation aussi bien qualitative que quantitative des performances des processus d'Extraction d'Information fondés sur cette base.

Le système collecte ou infère de chaque texte T_i à traiter, le plus possible de faits, de propositions et d'assertions afin de construire une base de connaissances textuelles (BCT) qui lui corresponde (*Text Knowledge Base* ou *TKB*). Chaque BCT_i représente le contenu informationnel du texte T_i et peut être utilisée par des outils de traitement d'information. Le processus d'extraction est réalisé par un outil interrogeant la BCT_i via des requêtes.

L'élaboration d'une BCT_i se déroule en plusieurs étapes : une phase de compréhension au niveau de la phrase, puis du texte et l'extension par apprentissage de la BCT_i ainsi que des sources permettant au système de construire des bases de connaissances textuelles.

La compréhension au niveau des phrases est fondée sur des connaissances conceptuelles et grammaticales ainsi que sur des schémas d'interprétation sémantique issus de ces deux types de connaissances. Le texte est analysé en utilisant une grammaire de dépendance fondée sur un lexique et chaque phrase est traduite en une hiérarchie de classes sémantiques de mots. Le lexique utilisé est constitué d'un lexique général contenant des mots indépendants du domaine ainsi que d'un lexique spécialisé couvrant les mots du domaine du texte (lexique de termes médicaux pour MEDSynDiKATe et informatiques pour ITSynDiKATe). Les connaissances conceptuelles sont établies grâce à un modèle du domaine exprimé dans un langage de représentation terminologique inspiré de KL-ONE [Woods & Schmolze 1992]. Cette ontologie est construite en combinant une ontologie exprimant des concepts généraux communs à l'ensemble des domaines des expérimentations et une ontologie spécifique au domaine dans lequel se situe le texte. L'interprétation sémantique consiste à déterminer les relations existant entre les instances des classes de concepts. Cette interprétation est réalisée par la mise en correspondance de l'ontologie du domaine avec le graphe de dépendance obtenu par l'analyse syntaxique. Les relations sont définies grâce aux entités lexicales communes aux deux modèles. Le processus produit un ensemble de connaissances sémantiques qui forment la base de la BCT_i.

Les problèmes d'incomplétude que pose un traitement uniquement au niveau de la phrase sont résolus par une phase de compréhension au niveau du texte. Les auteurs constatent qu'il n'est pas possible d'extraire des relations conceptuelles entre différentes entités appartenant à des phrases différentes sans passer par une analyse du texte dans son ensemble. Cette procédure consiste en la résolution d'anaphores (pronominales, nominales et fonctionnelles) en se fondant sur la théorie du centrage [16] [Grosz & al. 1995]. Les connaissances extraites lors de l'analyse au niveau du texte permettent également au système de corriger les incohérences ou invalidités présentes dans la BCT_i (comme par exemple des erreurs de conceptualisation liées à des ambiguïtés sémantiques comme l'homonymie).

La dernière étape est une phase d'apprentissage de connaissances nouvelles à partir du texte fondée sur des connaissances préalables sur le domaine du texte et des constructions grammaticales mettant en jeu des mots non référencés dans le lexique (entités lexicales inconnues). L'analyseur produit des graphes d'analyse de dépendances à partir de ces constructions grammaticales taxinomiques (apposition, phrases d'exemplification) ou agrégationnelles. L'interprétation conceptuelle de ces graphes entraîne la génération de concepts hypothétiques. Pour une entité lexicale inconnue, plusieurs concepts hypothétiques sont générés formant un espace d'hypothèses. Dans chaque espace d'hypothèse, un moteur fondé sur des labels de qualité linguistiques et conceptuels, estime la crédibilité de chaque concept hypothétique et détermine lequel correspond le mieux à l'entité. Les labels de qualité sont formés de patrons (de consistance, de justification mutuelle ou d'analogie) relatifs aux descriptions de concepts présents dans la base de connaissances ou d'autres hypothèses conceptuelles. Les nouveaux concepts viennent étendre la base de connaissances BCT_i du texte T_i analysé et enrichir les sources (grammaire, ontologie) qui permettent de créer des bases de connaissances textuelles pour d'autres textes du même domaine.

3.3.1.3 Projet MUMIS

Le projet MUMIS[17] (*Multi-Media Indexing and Searching Environment*) [Declerk & al. 2001] [De Jong & Westerveld 2001], mené sous la tutelle de la Société des Technologies de l'Information de l'Union Européenne[18], a pour volonté d'améliorer la recherche d'information dans des données multimédia grâce à un module d'Extraction d'Information fondé sur une ontologie. Le système traite d'enregistrements vidéo de matchs de football dans trois langues (allemand, anglais et néerlandais). Le but est d'indexer automatiquement ces documents multimédia afin de faciliter l'accès à des sous-parties (recherche de séquences de tirs au but par exemple). Dans MUMIS, chaque enregistrement vidéo s'accompagne de descriptions textuelles provenant d'articles de presse rendant compte du déroulement du match ainsi que de transcriptions du commentaire audio présent dans l'enregistrement. L'architecture du projet est découpée comme suit : un processus multilingue d'Extraction d'Information

[16] La théorie de centrage ou *Theory of centering* a été développée pour modéliser la cohérence locale d'un discours à travers la segmentation du discours en énonciations (*utterance*), la classification et le traitement des centres (*forward-looking center* et *backward-looking center*) de chaque énonciation. Les centres d'une énonciation correspondent aux entités du discours de cette énonciation [Poesio & al. 2000].

[17] `http://parlevink.cs.utwente.nl/projects/mumis/`

[18] *Information Society Technologies* (*IST*)

utilisant une ontologie codée en XML est appliqué sur les différents textes et extrait un ensemble de connaissances pour chaque match. Ces différentes connaissances sont ensuite encodées et fusionnées en un seul document transversal. Cette opération est réalisée afin d'obtenir une vision plus complète des événements qui se déroulent lors du match, notamment lorsque deux informations incomplètes sont extraites pour le même événement. Le document transversal est lié avec l'enregistrement vidéo du match grâce à des repères temporels et à une mise en relation des informations extraites avec la transcription du commentaire. Une interface permet aux utilisateurs de rechercher des séquences dans les enregistrements via la mise en correspondance de leurs requêtes avec la représentation de la connaissance contenue dans la vidéo.

Nous décrivons ici la première étape du processus[19], la phase d'Extraction d'Information qui a recours à une ontologie. Nous nous intéressons plus particulièrement à la construction de l'ontologie et sa mise en relation avec les textes. Sachant qu'il n'est pas possible de modéliser toute la connaissance possible d'un domaine donné, un ensemble de décisions doivent être prises lors de la construction d'une ontologie, ces décisions concernent le niveau de détail, la sélection des concepts et la définition des relations entre les concepts. Le principal critère de sélection retenu par les auteurs est la prise en considération de l'application pour laquelle l'ontologie doit être définie. Il s'agit de considérer non seulement le domaine dans lequel se place le système mais également la finalité du système, et de créer le modèle de la connaissance en fonction de cette finalité. Ici, l'application vise à rechercher des séquences dans un match de football, donc le modèle doit d'abord rendre compte des différents événements possibles dans un match de football comme les tirs au but, les coup-francs, les sorties de but ou les changements de joueurs. Les événements sont associés à des éléments comme les joueurs et les équipes ainsi que le moment de la partie où ils ont lieu. Ils forment les fondements de l'ontologie. Autour des événements sont construits d'autres concepts et relations modélisant les différents aspects d'un match de football. Certains aspects (concepts ou relations) qui peuvent paraître intéressants ou même évidents ne seront pas pris en compte si leur utilité pour l'application n'est pas avérée. Enfin, chaque concept défini est associé aux différentes expressions lexicales qui l'expriment dans les langues traitées via des listes de termes. L'ontologie créée apparaît alors comme un modèle pragmatique totalement tourné vers l'application plutôt que comme une conceptualisation se référant au sens philosophique du mot ontologie. Le modèle est encodé en utilisant le format XML.

Le module d'Extraction d'Information annote les textes avec des balises XML spécifiques à l'ontologie en utilisant des techniques de Traitement Automatique des Langues Naturelles. Le module procède à des analyses morphologiques et syntaxiques afin de détecter des patrons lexicaux qui correspondent à des événements particuliers. Les éléments liés conceptuellement à ces événements par des relations dans l'ontologie (sujet de l'événement, moment, etc.) sont annotés grâce à des processus[20] de reconnaissance des Entités Nommées

[19] Notre intérêt portant sur l'étude des moyens d'extraire de l'information via la construction et l'utilisation d'une ontologie, nous traiterons uniquement cet aspect du projet MUMIS. Nous ne détaillerons donc pas les processus de fusion, de mise en correspondance du texte annoté avec le signal vidéo ou de traitement des requêtes [Reidsma & al. 2003].

[20] Plusieurs de ces processus font appel à des méthodes développées au sein de GATE (*General Architecture for Text Engineering*) [Cunningham & al. 2003].

(pour extraire le nom des joueurs ou des équipes), de résolution de coréférences et d'analyse des anaphores [Cunningham 2002]. Les textes annotés sont ensuite fusionnés en un seul document afin de réunir toutes les informations sur un même événement. Le résultat est un document contenant des unités lexicales étiquetées par des balises XML exprimant des scénarios (sous-ensembles de concepts et de relations) décrits par l'ontologie et présents dans les documents sources.

3.3.1.4 Le système VULCAIN

Le système d'Extraction d'Information VULCAIN [Todirascu & al. 2002] a pour objectif d'extraire de l'information à partir de textes portant sur le domaine de la sécurité des systèmes informatiques. Le corpus traité est composé d'une grande quantité de messages électroniques portant sur le domaine concerné. Ces messages se caractérisent par quelques erreurs de syntaxe et d'orthographe et par des constructions syntaxiques spécifiques (notamment pour les noms de lieux, de personnes, d'organisations ou les fonctions).

La méthode consiste à identifier les termes et les concepts spécifiques au domaine en s'appuyant à la fois sur la syntaxe du texte et sur une ontologie du domaine. L'identification des termes est réalisée par une analyse syntaxique basée sur des grammaires d'arbres adjoints (grammaire TAG)[21]. L'analyse s'appuie sur une version modifiée de l'analyseur syntaxique LTAG (analyseur fondé sur des grammaires TAG lexicalisées[22]) [Lopez 1999] utilisant une combinaison de grammaires locales et de filtres statistiques portant sur les séquences grammaticales et fondés sur la quantité de mots pertinents qu'elles contiennent. L'ontologie du domaine est représentée par des logiques de descriptions (LD). Celles-ci possèdent une organisation hiérarchique et permettent de réaliser des inférences pour vérifier l'ontologie (validation des concepts candidats, mécanismes de vérification de la cohérence et de l'appartenance des instances aux classes de concepts).

À partir des résultats partiels de l'analyseur LTAG, des listes de groupes significatifs sont extraites, principalement des groupes nominaux de la forme Nom-Nom, Nom-Adjectif, Groupe Nominal-Préposition-Groupe Nominal. Ces entités identifiées (instances et catégories) sont liées à des éléments de l'ontologie par une interface syntaxe-sémantique qui se fonde sur des lexiques TAG spécifiques au domaine et décrits au format TAGML (mots-arbres élémentaires) ainsi que sur des lexiques sémantiques contenant des paires mot-

[21] Le modèle des grammaires d'arbres adjoints (*TAG* ou *Tree Adjoining Grammar*) [Joshi 1987] [Abeillé & Rambow 2000] tire son nom de l'utilisation d'arbres syntagmatiques (et non de règles de réécriture classiques) qui sont utilisés comme éléments atomiques. Ces arbres sont combinés à l'aide de deux opérations : l'adjonction, qui est l'opération spécifique aux TAG, et la substitution, qui est l'opération classique des grammaires hors contexte.

[22] Les grammaires d'arbres adjoints lexicalisés (*LTAG* pour *Lexicalized Tree Adjoining Grammar*) [Joshi & Schabes 1992], en intégrant la lexicalisation, permettent de mieux prendre en compte le contexte et de déterminer des propriétés des syntagmes dépendant d'éléments lexicaux. La lexicalisation permet d'éliminer certains traits d'unification qui dans d'autres formalismes ne font que gérer la sous-catégorisation et d'adopter de nouvelles stratégies en réduisant sensiblement la taille de la grammaire utilisée pour l'analyse syntaxique proprement dite. Dans les TAG lexicalisées, tout arbre élémentaire doit avoir au niveau de ses feuilles au moins un nœud terminal servant de lien avec les entités lexicales (ancre lexicale). La grammaire et le lexique ne forment ainsi qu'un seul ensemble. Des grammaires d'arbres adjoints lexicalisées ont été développées pour le français [Abeillé & Candito 2000] et pour l'anglais [XTAG-Group 1995].

concept. Les liens entre lexiques et ontologie sont ainsi représentés par des correspondances entre les arbres élémentaires et les structures conceptuelles associées aux concepts de l'ontologie du domaine.

Le modèle issu de la confrontation de l'ontologie préexistante avec l'analyse syntaxique du texte forme une nouvelle ontologie qui représente les informations contenues dans les textes. Ces informations peuvent ensuite être extraites et manipulées par un utilisateur grâce à des applications fondées sur les caractéristiques des logiques de description (règles de logique, inférences, raisonnement au niveau des instances).

3.3.1.5 Extraction de contenu informatif à partir d'Internet

Le développement et la démocratisation d'Internet ont incité à utiliser les pages Internet comme source d'informations textuelles. De nombreuses applications s'intéressant à la collecte d'informations à partir de pages Internet sont apparues [Eikvil 1999] comme par exemple les systèmes RAPIER [Califf 1999], WHISK [Soderland 1999], STALKER [Muslea & al. 1998] ou encore les systèmes commerciaux Junglee, Jango ou MySimon[23]). Parmi les travaux dans ce domaine, on trouve quelques systèmes d'Extraction d'Information qui ont recours à des ontologies pour étudier le contenu des pages (projets GETESS et CROSSMARC).

3.3.1.5.1 Projet GETESS

Le projet GETESS (*GErman Text Exploitation and Search System*)[24] [Staab & al. 1999] cherche à exploiter des pages Internet en langue allemande pour récolter des informations concernant le tourisme et les présenter de manière intuitive et claire. Il fait appel à un ensemble de méthodes sémantiques et de processus de Traitement Automatique des Langues Naturelles et intègre un système d'Extraction d'Information basé sur une ontologie construite à l'aide d'un mécanisme d'amorçage [Maedche & al. 2002].

3.3.1.5.2 Projet CROSSMARC

Le projet CROSSMARC (*Cross-lingual multi-agent retail comparison*)[25] [Hachey & al. 2003] [Valarakos & al. 2003] de l'Union Européenne a pour but de rechercher et d'identifier des pages Internet contenant des descriptions de produits puis d'en extraire des informations sur ces produits. Le projet se fonde sur la définition d'une ontologie et le développement d'outils de gestion d'informations utilisant cette ontologie. CROSSMARC intègre une dimension multilingue puisque l'objectif est de pouvoir traiter des pages Internet écrites dans différentes langues.

[23] `http://www.mysimon.com`

[24] `http://www.getess.de`

[25] `http://iit.demokritos.gr/skel/crossmarc/`

L'ontologie est définie de manière assez flexible pour être appliquée à différents domaines et langues sans pour autant devoir changer sa structure générale. Elle est facilement modifiable et adaptable. Pour répondre à ces attentes, elle est construite sur trois niveaux : un niveau méta-conceptuel, un niveau conceptuel et un niveau instance.

- Le niveau méta-conceptuel est le niveau le plus élevé de l'ontologie. Il est indépendant du domaine étudié et détermine l'architecture de l'ontologie, c'est-à-dire les structures qui seront respectées par tous les éléments du modèle. Ce niveau définit également la structure des formulaires qui seront utilisés dans le processus d'Extraction d'Information ;

- Le niveau conceptuel est composé des concepts ayant trait au domaine particulier dans lequel se placent les textes traités. La représentation interne de ces concepts ainsi que les relations inter-concepts suivent la structure définie au niveau méta-conceptuel. À chaque concept est associé un identifiant numérique unique appelé *onto-référence*. Le niveau conceptuel définit la sémantique du domaine considéré et permet de construire des patrons d'extraction spécifiques au domaine ;

- Le niveau instance représente les particularités du domaine décrit par les textes. Il inclut des instances de concepts qui correspondent à des valeurs normalisées de chaque particularité ainsi que des instances lexicales des concepts dans chacune des langues (représentation lexicale usuelle du concept et synonymes). Chaque instance est unique et possède un seul identifiant nommé *onto-value*. Pour assurer le caractère multilingue de l'ontologie, le système utilise des lexiques spécifiques au domaine pour chacune des langues traitées.

Le processus d'Extraction d'Information développé dans CROSSMARC utilise une version XML de l'ontologie. Il commence par une phase de reconnaissance des entités nommées grâce à des index issus des lexiques et de l'ontologie. Ensuite, l'application procède à une phase d'extraction de faits en comparant les entrées normalisées de l'ontologie et leurs synonymes avec les entités lexicales présentes dans le texte. Si une correspondance est trouvée, le texte est annoté en étiquetant le fait par l'identifiant du nœud de l'ontologie correspondant. Les informations extraites (portant sur des produits) sont ensuite collectées via les étiquettes et stockées dans une base de données dont la structure est déterminée par les formulaires d'extraction.

3.3.2 Analyse des méthodes

Les méthodes présentées dans la section précédente cherchent à produire une modélisation des informations du texte en mettant en relation son contenu avec un modèle prédéfini. Il s'agit d'exploiter les entités lexicales et syntaxiques du texte, soit en les liant avec une ontologie construite préalablement (comme dans le système VULCAIN ou les projets MUMIS et CROSSMARC), soit en construisant à partir d'elles un modèle sous un formalisme particulier (grammaire, hiérarchie de concepts) et en unifiant le modèle avec une ontologie préalable pour obtenir un modèle complet et fidèle des connaissances du texte (systèmes LaSIE et SynDiKATe). L'extraction des informations est réalisée à partir de celles qui sont modélisées.

Ces approches produisent des résultats intéressants mais, à l'instar des méthodes classiques d'Extraction d'Information décrites dans le chapitre 2, elles ne s'avèrent pas adaptées aux Notes de Communication Orale en raison de leur recours à des techniques de Traitement Automatique des Langues. En effet, au cœur de chacune de ces approches se trouvent des modules de reconnaissance d'entités nommées, de résolution d'anaphores ou de coréférences (MUMIS, SynDikate) et surtout des outils d'analyse syntaxique fondés sur des patrons syntaxiques ou des grammaires (grammaire de dépendance dans SynDiKATe ou grammaire d'arbres adjoints lexicalisés dans VULCAIN). Ainsi la qualité du modèle et les performances des systèmes sont extrêmement liées aux résultats de l'utilisation de ces techniques.

Cependant, ces approches ont abouti au développement de méthodes de création et d'utilisation d'ontologies en relation avec des textes qui présentent des aspects intéressants pour la réalisation d'un processus d'Extraction d'Information adapté aux Notes de Communication Orale : le recours à des connaissances externes aux textes et la prise en compte du but de l'application lors de la construction de l'ontologie.

3.3.2.1 Recourir à des connaissances externes aux textes

Les travaux menés dans le cadre des systèmes LaSIE, CROSSMARC et SynDiKATe présentent l'intérêt de recourir à des connaissances externes aux textes pour l'élaboration d'une ontologie modélisant les informations qu'ils contiennent (recours à des lexiques multilingues spécialisés dans CROSSMARC, à des lexiques généraux et spécialisés dans SynDiKATe et à une ontologie élaborée à partir des formulaires d'extraction dans LaSIE). Il s'agit de s'appuyer sur ces connaissances externes pour élaborer un modèle externe aux textes (conceptualisation des termes des lexiques dans SynDiKATe et CROSSMARC, *world model* dans LaSIE) qui servira de base pour la construction de l'ontologie modélisant les connaissances du texte ou qui sera combiné à une ontologie générée à partir du texte pour former une représentation complète de ses connaissances.

Cette idée de s'appuyer sur des connaissances externes aux corpus pour construire l'ontologie est intéressante du point de vue des Notes de Communication Orale. En effet les particularités linguistiques de ces textes peuvent rendre problématique le recours unique à l'étude des textes pour générer un modèle de ses connaissances. Aussi, utiliser une source de connaissances externe au texte possède l'avantage de permettre de s'abstraire, du moins partiellement, d'une analyse textuelle aux résultats incertains, pour la construction du modèle. Une telle connaissance externe peut être apportée par des ressources linguistiques (notamment des lexiques spécialisés) ou des experts. Cette dernière affirmation est particulièrement vraie pour des corpus issus d'entreprises ou d'institutions dans lesquelles se trouvent des spécialistes à même de connaître les concepts du domaine, et avec lesquels il est possible de travailler pour élaborer un modèle des connaissances des textes à traiter.

3.3.2.2 Prendre en compte le but

En se plaçant dans le cadre du traitement d'un ensemble de textes d'un domaine particulier, les connaissances de ce domaine (décrites dans une ontologie générale ou de domaine, ou bien exprimées par un expert) ne seront pas toutes présentes dans chaque texte étudié ni même dans chacun des corpus traités car ceux-ci ne forment qu'une sous-partie du domaine. Lors de la construction d'une ontologie d'application pour un corpus de textes particuliers, seule une partie des connaissances du domaine doit être retenue : un ensemble composé des éléments pertinents vis-à-vis des textes et des traitements à réaliser. Se pose alors la question suivante : comment sélectionner les éléments conceptuels à retenir ?

Les équipes de développement de LaSIE et de MUMIS proposent des réponses fondées sur la prise en compte du but de l'application dans la construction de l'ontologie. Dans LaSIE le *modèle du monde* qui sert de base à l'ontologie finale (*modèle du discours*) est défini par une hiérarchie de concepts construite sur la base des formulaires à remplir. Les informations recherchées forment ainsi une partie des concepts de l'ontologie. Le *modèle du monde* est ensuite complété par des concepts liés hiérarchiquement aux concepts issus du formulaire. La même idée est présente dans MUMIS. Ici aussi les informations à rechercher forment la base de l'ontologie. Les concepts et les relations choisis sont ceux qui représentent les éléments visés par l'application. D'autres concepts et relations, représentant le domaine et liés au but poursuivi par le système, viennent enrichir l'ontologie alors que seront mis de côté des concepts et relations qui n'apparaissent pas pertinents vis-à-vis de ces buts (même si leur existence dans le domaine est évidente).

La prise en compte du but apparaît comme essentielle pour la construction d'une ontologie destinée à faire partie d'un processus d'Extraction d'Information. Elle présente les avantages suivants :

- L'assurance de représenter les différents éléments à extraire et de pouvoir les relier au texte via un ensemble de concepts et de relations ;
- La limitation de la taille de l'ontologie puisque l'on ne cherche pas à modéliser toutes les connaissances contenues dans les textes mais seulement celles liées avec les informations à rechercher. Cette limitation rend l'ontologie plus facile à construire et à manipuler ;
- L'assurance de posséder un point de vue unique sur les textes à traiter (celui des buts à atteindre) ce qui évite d'éventuelles ambiguïtés ou contre-sens sur les termes ou les concepts présents dans les textes ;
- La possibilité d'adapter le processus pour différents domaines car les fondements de la modélisation ne dépendent pas d'un domaine particulier ;
- La prise de distance avec la surface des textes lors de la construction de l'ontologie permet de limiter l'impact de problèmes orthographiques et/ou syntaxiques dans les textes traités.

3.4 Un modèle guidé par le but

Les caractéristiques des Notes de Communication Orale nous ont amené à porter nos recherches sur l'élaboration d'une méthode d'extraction fondée sur une modélisation des connaissances à travers une ontologie. Notre méthode **MEGET** (Méthode Générique d'Extraction d'informations à partir de Textes) s'appuie ainsi sur la construction d'une ontologie (appelée ***ontologie d'extraction*** ou ***OE***) qui sera utilisée comme base de connaissance d'un système d'Extraction d'Information. Le système, nommé **SYGET** (Système Générique d'Extraction d'informations à partir de Textes), extrait des informations d'un texte en y repérant les instances des éléments de l'ontologie via un étiquetage des textes (à l'instar des projets MUMIS et CROSSMARC).

La méthodologie de construction de l'ontologie d'extraction est adaptée aux particularités des Notes de Communication Orale. Nous l'avons définie après une étude des techniques utilisées dans les différentes méthodes de construction d'ontologie (cf. section 3.2.4) et dans les systèmes d'Extraction d'Information fondés sur une ontologie (cf. section 3.3.2).

L'idée principale sur laquelle se fonde notre méthodologie est que la prise en compte du but doit être à la base de la méthode de construction d'une ontologie élaborée dans un objectif précis. Il est inutile de chercher à rendre compte systématiquement de tous les énoncés, seules les connaissances utiles à l'objectif exprimé sont intéressantes et doivent être modélisées. Cette idée s'inscrit dans le courant des méthodes descendantes de construction d'ontologie (KADS ou méthodes dirigées par un modèle) [Schreiber & al. 1993] des méthodes orientées vers la tâche à réaliser. La notion d'ontologie utilisée ici doit être vue comme une description structurée de concepts et de relations, représentés formellement et choisis pour leur adéquation avec l'objectif de l'application fondée sur le modèle [Aussenac-Gilles & Condamines 2001]. À ce titre, « *une ontologie fonctionne comme un cadre théorique du domaine construit en fonction du problème traité* » [Bachimont 2001].

Dans **MEGET**, nous avons appliqué cette idée de la prise en compte du but au principe de l'Extraction d'Information. L'objectif de l'Extraction d'Information est bien établi, il s'agit de récupérer dans un ou plusieurs textes un ensemble d'informations précises définies en amont. En conséquence seules les connaissances liées avec les informations à rechercher doivent être modélisées dans l'ontologie d'extraction. Cette ontologie est ainsi définie comme un ensemble structuré de concepts et de relations décrivant de manière précise les connaissances à rechercher. Une telle ontologie permet d'identifier ces connaissances dans les textes grâce aux représentations lexicales des concepts et des relations de l'ontologie. Le système d'extraction **SYGET** exploite l'ontologie d'extraction en suivant ce principe.

Les caractéristiques linguistiques des Notes de Communication Orale ne permettent pas de se fonder uniquement sur le contenu du texte pour élaborer un modèle conceptuel. En conséquence, il est nécessaire de s'appuyer sur des connaissances externes indépendantes des textes pour construire l'ontologie d'extraction. Ces connaissances externes doivent permettre de modéliser correctement et précisément les informations à extraire. Une bonne solution pour acquérir de telles connaissances est de recourir à des experts appartenant au domaine concerné et capables d'exprimer de manière assez complète et précise les informations à rechercher. Néanmoins le seul recours aux experts n'est pas suffisant, étant établi que des

experts ne sont pas en mesure de verbaliser explicitement et complètement un ensemble de termes qui seraient la traduction lexicale d'un système conceptuel dont ces mêmes experts sont pourtant à la base [Bourigault & Jacquemin 2000]. Aussi une étude du contenu des textes apparaît nécessaire pour déterminer les représentations lexicales des concepts et permettre de lier le modèle au texte pour y repérer les instances des concepts. Cette étude peut être effectuée grâce à une analyse terminologique mêlant l'application d'un processus d'extraction de terminologie (à condition que celui-ci ne soit pas fondé sur des outils linguistiques incompatibles avec les Notes de Communication Orale, cf. chapitre 4) et la consultation de ressources linguistiques externes aux textes (lexiques).

En se fondant sur les principes précédents, notre approche (résumée par la figure 3.1) associe deux modèles pour former l'ontologie d'extraction. Le premier modèle que nous appelons ***ontologie des besoins*** (ou ***OB***) est construit à partir de l'expression des informations à rechercher. Le second modèle, intitulé ***ontologie des termes*** (ou ***OT***), est une ressource termino-ontologique correspondant à une conceptualisation des termes présents dans les textes à traiter et intéressants du point de vue de l'extraction (ontologie issue des termes). La méthode de construction se découpe en deux phases : premièrement l'élaboration de l'ontologie des besoins grâce à un travail coopératif avec des experts du domaine et deuxièmement une étude termino-ontologique qui aboutit à la création de l'ontologie des termes. Les deux ontologies sont unifiées pour obtenir l'ontologie d'extraction.

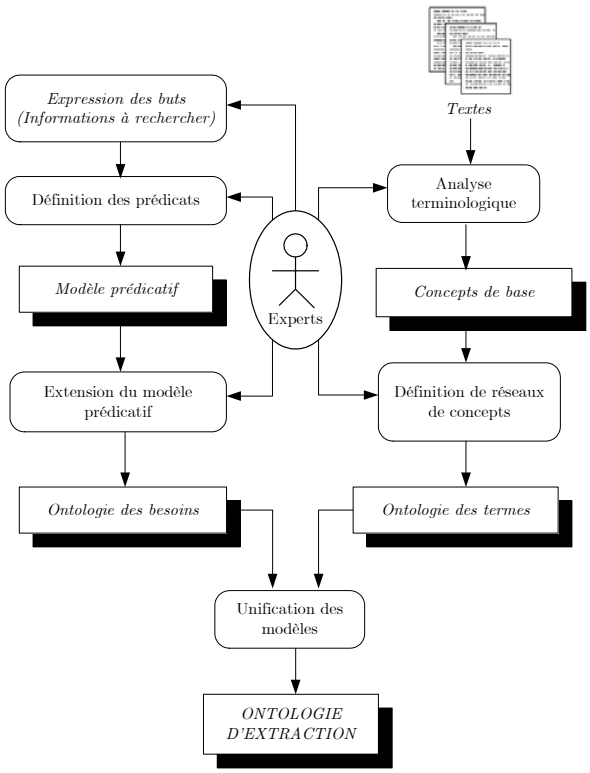

Figure 3.1 : Méthodologie de construction de l'ontologie d'extraction

Chapitre 4

Extraction de terminologie et Notes de Communication Orale

Présentation

L'identification des informations décrites par les concepts d'une ontologie nécessite d'établir un lien entre leur description conceptuelle et leur description dans les textes. Il s'agit ainsi d'expliciter les différentes représentations linguistiques des concepts de l'ontologie par la détermination d'un ensemble de termes, les termes constituant la manifestation linguistique des objets réels ou immatériels que l'on nomme concepts [Sager 1990].

Dans ce chapitre, nous discutons d'abord de la notion de terme (section 4.1), puis décrivons les principaux outils d'acquisition automatique de terminologie. Ces outils sont séparés en trois catégories : l'approche linguistique regroupant les méthodes exploitant les caractéristiques linguistiques du texte (section 4.2), l'approche statistique qui rassemble les outils fondés sur une analyse statistique du texte en étudiant principalement le phénomène d'occurrence (section 4.3) et l'approche hybride combinant les deux approches précédentes (section 4.4). Après avoir décrit l'ensemble de ces outils, nous présentons les adéquations ou inadéquations de leurs approches avec les caractéristiques linguistiques des Notes de Communication Orale (section 4.5). Le résultat de cette étude nous a permis de retenir le système d'extraction de terminologie qui sera utilisé lors de l'étude termino-ontologique (cf. chapitre 6).

4.1 La notion de terme

Dans la conception classique de la terminologie, théorisée par Eugène Wüster [Wüster 1979] dans sa *Théorie Générale de la Terminologie* [Wüster 1976], le terme désigne une notion (et par conséquent un concept) de façon univoque, mono-référentielle et non contextuelle. Il est un symbole conventionnel représentant une notion définie dans un certain domaine du savoir, c'est-à-dire le représentant linguistique d'un concept dans un domaine de connaissance [Felber 1987]. Une terminologie se veut alors une représentation parfaite d'un système conceptuel sous-jacent, le terme devenant une simple étiquette non ambiguë d'une notion au sein d'un domaine structuré en une taxinomie [Rousselot & Frath 2002]. Cette conception de la terminologie est maintenant rejetée par les chercheurs qui considèrent que le terme est un mot de la langue et que le modèle conceptuel du domaine est une construction personnelle reflétant un point de vue sur les connaissances du domaine.

Le courant du renouvellement théorique de la terminologie [Rastier 1995] s'écarte de la théorie de Wüster et s'appuie sur le constat « *qu'étant donné un domaine d'activité, il n'y a pas une seule terminologie représentant le savoir d'un domaine mais autant de terminologies que d'applications dans lesquelles ces terminologies sont utilisées* » [Bourigault & Jacquemin 2000]. Du point de vue des unités retenues et de leur description, les terminologies peuvent nettement différer selon l'application. Le terme est vu comme le résultat d'une analyse termino-conceptuelle dans laquelle une entité (mot, groupe de mot ou autre unité lexicale) issue d'un corpus et appelée *candidat-terme*, n'acquiert le statut de terme[1] que par décision [Bourigault & al. 2004]. Cette décision doit être prise par l'analyste qui définit son propre référentiel de décision en fonction des objectifs de l'application (par exemple, dans le cas de la conception d'un modèle conceptuel, cette tâche sera allouée à l'ontologue aidé éventuellement d'un ou plusieurs experts : experts du domaine, terminologues). Il s'agit de ne retenir comme termes que les unités lexicales présentant des caractéristiques propres au domaine et utiles à l'application pour laquelle la ressource terminologique est constituée. La construction de la terminologie se doit d'être pertinente à la fois vis-à-vis du corpus et de l'application.

Une analyse terminologique consiste donc d'abord à extraire automatiquement du corpus un ensemble de candidats-termes qui seront ensuite fournis à un analyste chargé de déterminer lesquels doivent être retenus comme termes.

4.2 Approche linguistique

4.2.1 TERMINO

Les premiers travaux en acquisition de terminologie correspondent à l'élaboration du système TERMINO à la fin des années 1980 [David & Plante 1990]. TERMINO (devenu aujourd'hui le logiciel NOMINO[2]) extrait des candidats-termes (appelés ici *synapsies*[3]) à

[1] D'après la norme ISO 1087 1990 : 5 [ISO 1969], un terme peut être constitué d'un ou de plusieurs mots (terme simple ou terme complexe) et même de symboles (chiffres, symboles non-alphanumériques).

[2] `http://www.ling.uqam.ca/nomino/`

travers le repérage des syntagmes nominaux dans le corpus. Après une étape de prétraitements, le système procède à une analyse morphologique puis à une analyse syntaxique des textes fondée sur un étiquetage selon une grammaire du syntagme nominal de type X-barre [Jackendoff 1977]. Les synapsies sont identifiées à partir des différentes expansions possibles de chaque nom. Les expansions sont détectées grâce à la structure de syntagmes nominaux issus de l'analyse. Les noms et syntagmes ainsi obtenus et jugés valides sont fournis sous la forme d'une liste triée de candidats-termes (alphabétiquement ou par fréquence dans le corpus).

4.2.2 LEXTER

LEXTER est un logiciel d'extraction terminologique destiné à l'acquisition de connaissances à partir de textes techniques. Il extrait des candidats-termes à partir d'un corpus préalablement étiqueté et désambiguïsé [Bourigault 1994]. L'idée à la base de ce système est le constat qu'un certain nombre de constituants du texte (ponctuation ou éléments du discours comme les verbes, les pronoms ou les adverbes) ne peuvent faire partie d'un terme et forment les frontières des termes dans le corpus. À partir du corpus étiqueté LEXTER effectue une analyse morpho-syntaxique qui lui permet de repérer et d'analyser des syntagmes nominaux susceptibles d'être des termes. Le résultat est un ensemble de candidats-termes organisé en réseau grammatical.

L'acquisition de candidats-termes se déroule en plusieurs étapes. D'abord une analyse morphologique assigne aux mots de la phrase une étiquette grammaticale, la ponctuation est également étiquetée. Les groupes nominaux maximaux sont identifiés par le repérage de leurs frontières syntaxiques (verbes conjugués, pronoms, conjonctions, etc.). Il s'agit d'une analyse en négatif du corpus car le système repère les éléments syntaxiques ne pouvant être des constituants de termes pour délimiter les syntagmes nominaux. Un module de décomposition analyse ensuite de manière récursive les syntagmes nominaux maximaux en paires tête-expansion. Les syntagmes nominaux maximaux et leurs constituants sont renvoyés comme candidats-termes.

Un module de structuration organise les candidats-termes en un réseau terminologique en établissant des liens entre chaque candidat-terme et les autres candidats-termes dans lesquels il est en position tête ou expansion. LEXTER procède ainsi à une première structuration des termes. Dans un second temps des techniques d'analyse statistique réalisent un filtrage, un typage et une classification des candidats-termes.

Cette approche a recours à des techniques d'apprentissage endogène sur corpus dans les phases d'extraction de syntagmes nominaux maximaux et de décomposition pour résoudre les problèmes d'ambiguïtés de rattachement prépositionnels et adjectivaux au sein des groupes nominaux.

[3] D'après la définition d'Émile Benveniste « *Une synapsie consiste en un groupe entier de lexèmes, reliés par divers procédés, et formant une désignation constante et spécifique.* » [Benveniste 1966]

4.2.3 SYNTEX

Le logiciel SYNTEX [Bourigault & Fabre 2000] est un analyseur syntaxique de corpus en français ou en anglais. Cet outil renvoie en résultat un ensemble de mots et syntagmes et est utilisable comme extracteur de terminologie. On trouve des exemples d'une telle utilisation de SYNTEX dans les projets de construction de ressources terminologiques et ontologiques VERRE, REA et DROIT [Bourigault & al. 2004].

SYNTEX identifie à partir d'un corpus étiqueté des noms, des verbes, des adjectifs et des syntagmes nominaux, verbaux et adjectivaux (groupe de mots dont la tête syntaxique est respectivement un nom, un verbe ou un adjectif). Ces éléments forment un réseau de dépendance syntaxique dit *réseau terminologique* dans lequel chaque syntagme est relié d'une part à sa tête (lien T) et d'autre part à ses expansions (lien E). Les éléments du réseau forment l'ensemble des candidats-termes. Pour chaque candidat-terme, SYNTEX fournit sa fréquence dans le corpus ainsi que sa productivité en tête et en expansion, c'est-à-dire son nombre d'apparitions en tant que tête (respectivement en tant qu'expansion) d'autres candidats-termes.

4.3 Approche statistique

4.3.1 MANTEX

MANTEX [Rousselot & Frath 2002] [Oueslati 1999] est un système d'aide à l'extraction de terminologie à partir de textes non étiquetés. Ce système est désormais intégré à la station LIKES (LInguistic Knowledge Engineering Station[4]). MANTEX est fondé sur la méthode des segments répétés de Ludovic Lebart et André Salem [Lebart & Salem 1994] : « *toutes les suites d'occurrences non séparées par un délimiteur de séquence sont des occurrences de segments ou de polyformes*[5]*. Les segments dont la fréquence est supérieure ou égale à deux dans le corpus sont des segments répétés dans le corpus* ». Dans MANTEX les délimiteurs sont les signes de ponctuation ou les espaces ainsi que les verbes, les pronoms (personnels, relatifs, etc.) et certains adverbes.

Le système procède à une indexation de l'ensemble des mots du texte en leur attribuant un code correspondant à leur position dans le corpus. Ensuite il effectue un repérage de tous les segments répétés dans une fenêtre de un à dix mots en se limitant à la même phrase. Lors de cette phase les redondances sont éliminées en supprimant les segments inclus dans d'autres avec le même nombre d'occurrences [Frath & al. 2000].

À ce stade MANTEX a extrait un grand nombre de segments dont certains sont incorrects. L'ensemble de ces segments est ensuite filtré pour éliminer les segments indésirables et ne conserver que ceux qui seront retenus comme candidats-termes. Le système a recours à deux filtres : un filtre grammatical et un filtre coupant. Le filtre

[4] Une version téléchargeable de la station LIKES est disponible sur le site du LIIA de Strasbourg : `http://www- http://www-ensais.u-strasbg.fr/liia/LIIA_Products_Installers/install.htm`

[5] Un polyforme correspond à une séquence polylexicale, c'est-à-dire un groupement d'au moins deux lexèmes.

grammatical élimine les segments commençant ou finissant par un mot appartenant à une liste de mots grammaticaux (articles, déterminants, conjonctions, propositions, certains adverbes). Les mots du filtre grammatical peuvent être présents à l'intérieur d'un segment. Le filtre coupant permet de découper ou de supprimer les segments comportant certains mots comme des verbes, des entités nommées, des nombres ou des mots non spécifiques à la terminologie du domaine. Les segments recherchés étant des candidats-termes donc désignant des notions décontextualisables, ceux qui contiennent des mots liant le segment à une situation particulière comme les possessifs ou les démonstratifs par exemple sont également éliminées avec ce filtre. Le filtre permet aussi de générer des segments en découpant ceux comportant des conjonctions de coordinations (segments décrivant plusieurs termes). La liste des mots du filtre coupant fournie avec le système est minimale et générale, mais peut être aisément adaptée et complétée par l'utilisateur de MANTEX en fonction des spécificités des corpus traités. À la différence des mots du filtre grammatical, ceux du filtre coupant ne peuvent plus être présents dans un segment après application de ce filtre.

L'ensemble de candidats-termes est obtenu après une phase de regroupement des segments restants, en couples « *premier mot – dernier mot* ». Ce processus permet de regrouper les segments répétés qui se différencient seulement par la morphologie (utilisation d'une méthode de lemmatisation sommaire) et d'identifier des segments non répétés proches de ceux qui ont été répétés mais qui en diffèrent morphologiquement ou syntaxiquement.

Dans MANTEX, la représentation des candidats-termes repose sur une factorisation des facteurs gauches communs appelés têtes (représentation en arbre). Chaque terme est ainsi visuellement décomposé en une tête et un ensemble d'extensions.

4.3.2 ANA

Le système ANA (Acquisition Naturelle Automatique) [Enguehard 1992, 1993] est un outil d'extraction de terminologie dont la méthodologie est inspirée par l'apprentissage de la langue naturelle chez les enfants (apprentissage par association des sons avec des perceptions puis par induction et généralisation de ces associations). Cette approche évite toute analyse linguistique des corpus traités en se fondant sur l'idée que les textes sur lesquels sont réalisées les acquisitions terminologiques ne sont pas obligatoirement de bonne qualité et peuvent ainsi comporter des séquences syntaxiques incorrectes faussant les résultats des analyseurs syntaxiques. En conséquence ANA est un système essentiellement fondé sur des techniques statistiques. Il se déroule en deux phases successives (*familiarisation* et *découverte*) imitant l'apprentissage humain et s'appuie principalement sur l'observation de patrons de mots fréquents et l'égalité souple entre termes [Enguehard 2000].

Le processus commence par une phase, dite de familiarisation, qui génère des listes de mots et de termes qui seront utilisées dans la phase suivante (phase de découverte). Une liste de mots fonctionnels[6] est d'abord produite en analysant plusieurs corpus de domaines différents et en ne retenant que les mots les plus fréquents de l'ensemble des corpus. Cette

[6] Voir la note n°18 du chapitre 2 (page 46).

méthode se fonde sur le principe que les mots fonctionnels sont utilisés dans tous les domaines et ce de manière massive. Elle permet d'obtenir une cinquantaine de mots couvrant la grande partie les mots fonctionnels existants. Le système détermine ensuite un ensemble de mots (termes simples) représentant des concepts du domaine et formant l'amorce du système (par exemple "COEUR", "CUVE", "STRUCTURE" et "REACTEUR" [7] dans un corpus traitant de centrales nucléaires). Il s'agit des dix à vingt mots non fonctionnels les plus fréquents dans le corpus. À partir de cette liste est établi un premier ensemble de termes complexes en détectant les cooccurrences [8] de mots de l'amorce, éventuellement séparés par un mot fonctionnel et dont la fréquence dans le corpus est supérieure à un seuil fixé au préalable. Ce processus se fonde sur le postulat que les cooccurrences fréquentes sont significatives ("COEUR DE REACTEUR " est un exemple de terme complexe). Les mots fonctionnels mis en jeu dans des termes complexes sont réunis dans une liste de schémas lexicaux ("*du*", "*de*", "*de la*", "*de ce*", etc.). Les nouveaux termes viennent enrichir l'amorce.

Le module de découverte se déroule de manière incrémentielle à partir de la liste amorce et de celle des schémas lexicaux. Le module détermine de nouveaux termes en repérant deux types de cooccurrences : les cooccurrences d'un terme et d'un mot séparés par un schéma lexical (par exemple "CUVE du barillet"), et les cooccurrences d'un terme et d'un mot sans présence d'un schéma lexical (par exemple "STRUCTURES internes"). Les cooccurrences ayant une fréquence jugée acceptable sont retenues comme termes et ajoutées à l'amorce. Le processus est relancé sur le corpus à partir de cette nouvelle amorce et ainsi de suite jusqu'à ce qu'aucun candidat-terme nouveau n'apparaisse lors d'un cycle de traitement. Le système *apprend* de cette manière les candidats-termes du corpus.

Grâce au principe de l'égalité souple (ou « *flexible-equality*») [Enguehard 2000], ANA est capable de repérer des termes rencontrés sous différentes graphies (flexions, fautes orthographiques, etc.) lors des deux phases du système.

Les candidats-termes extraits par ANA sont présentés par une liste de candidats-termes accompagnés de leur variantes relevées dans le corpus ou de couples « candidats-termes/fréquence dans le corpus », ou bien par un réseau sémantique décomposant chaque candidat-terme complexe selon ses composants (par exemple "COEUR DE REACTEUR " avec "COEUR" d'une part et "REACTEUR" d'autre part).

[7] Les mots repérés comme candidats-termes sont écrits en lettres capitales.

[8] En reprenant les définitions données par Pierre Frath dans sa thèse [Frath 1997], la cooccurrence est définie comme « *la co-présence de deux mots dans une fenêtre, c'est-à-dire dans un contexte gauche et droit d'une taille définie* », la collocation est définie comme « *la coprésence juxtaposée de deux ou plusieurs mots, c'est-à-dire une cooccurrence particulière. Ainsi "**pomme**" et "**terre**" sont des cooccurrents, et "**pomme de terre**" est une collocation* ».

4.4 Approche hybride

4.4.1 ACABIT

Le système ACABIT [Daille 1994, 1996] travaille sur un corpus préalablement étiqueté et désambiguïsé. Il extrait automatiquement des candidats-termes par une analyse syntaxique du corpus suivie de traitements statistiques filtrant les résultats de l'analyse.

À partir d'un corpus lemmatisé et étiqueté par un assignateur de catégories stochastiques, ACABIT réalise une première acquisition de terminologie grâce à l'utilisation d'un ensemble de transducteurs, le résultat est une liste de candidats-termes contenant des termes de base (termes composés de deux unités lexicales pleines) correspondant à des schémas morphosyntaxiques simples (*N1 N2, N Adj, N1 Prep N2, N1 Prep Dét N2*) et des termes plus complexes (de longueur supérieure à 2) obtenus à partir des termes de base par des opérations d'insertion et de juxtaposition en s'inspirant des travaux de Christian Jacquemin sur les compositions et modification des noms composés terminologiques [Jacquemin 1991]. Cette phase linguistique intègre la prise en compte des variations sur les termes complexes (variations flexionnelles et syntaxiques, variations morphosyntaxiques et anaphoriques) [Daille 2001, 2003].

La liste de candidats-termes obtenue par la phase précédente est ensuite filtrée au moyen de mesures statistiques. Le recours à la simple fréquence des termes dans le corpus apparaissant insuffisante pour détecter les termes (risque d'oubli de termes significatifs du domaine mais peu fréquents, ou conservation de syntagmes libres répétitifs), Béatrice Daille retient le calcul statistique du *coefficient de vraisemblance* [Dunning 1993]. Ce modèle statistique fournit des listes de termes moins bruitées et plus complètes que celles obtenues par un calcul de fréquence. Les candidats-termes sont sélectionnés et classés en fonction de la valeur qui leur est attribuée par le coefficient de vraisemblance, les plus fortes valeurs désignant les candidats-termes les plus caractéristiques du domaine.

4.4.2 XTRACT

XTRACT [Smadja 1993] n'est pas un logiciel dédié spécifiquement à la terminologie, il s'agit d'abord d'un extracteur de collocations. Il trouve néanmoins sa place dans cette description car il peut être utilisé en acquisition de terminologie.

Les collocations sont détectées de manière statistique en se fondant sur le principe que les mots d'une collocation apparaissent ensemble plus fréquemment que lorsque le seul hasard intervient (mesure de l'Information Mutuelle [Church & Hanks 1990]). Elles sont ensuite classées et filtrées selon des critères syntaxiques.

Le système est fondé sur un module d'extraction de collocations binaires qui extrait les couples de mots qui se rencontrent à des distances fixes l'un de l'autre de manière plus fréquente que par hasard. Le module commence par repérer les co-occurrents d'un mot donné dans une fenêtre de dix mots (cinq à gauche et cinq à droite) dans une même phrase en notant la place du co-occurrent par rapport au mot cible. Le système sélectionne les co-

occurrents dont la fréquence dépasse de manière statistiquement significative la fréquence due au hasard. Un seuil numérique (80%) est défini *a priori* pour estimer qu'une relation entre deux éléments est significative. Les collocations ainsi extraites sont ensuite étendues afin de produire des collocations de plus de deux mots. Cette extension est réalisée en utilisant de manière récursive la méthode précédente sur les phrases comprenant des collocations significatives. Les collocations collectées correspondant à des formes linguistiques différentes, un analyseur syntaxique (analyseur CASS [Abney 1990]) les classe en catégories syntaxiques (associations verbe et objets typiques, collocations prédicatives verbe support et nom prédicatif) et syntagmes figés. Un filtrage des collocations en fonction de leur catégorie est ensuite effectué : si une collocation correspond à une forme syntaxique spécifiée par l'utilisateur alors elle est validée, sinon elle est rejetée. Les principales classes de collocations extraites par XTRACT sont les collocations prédicatives (verbe support et nom prédicatif comme "*make decision*"), les associations verbe et objet typique ("*reach an agreement*") et les syntagmes figés ("*stock market*").

Dans XTRACT le filtrage linguistique est effectué en sortie de l'extraction statistique. Ce processus peut être vu comme une construction miroir d'ACABIT dans laquelle l'enchaînement des méthodes statistique et linguistique est réalisé dans le sens inverse.

4.4.3 TERMS

Le système TERMS [Justeson & Katz 1995] se fonde sur la double idée que les termes sont répétés dans un document technique plus fréquemment que les syntagmes non terminologiques et qu'ils possèdent une structure et des variantes différentes de celles des syntagmes non terminologiques. La prise en compte de la répétition comme facteur déterminant de la reconnaissance de termes rapproche la méthode de TERMS des approches statistiques fondées sur une étude des fréquences des séquences. L'extraction des termes est réalisée par la reconnaissance de patrons syntaxiques à partir d'un corpus étiqueté. Les patrons sont écrits à partir de l'étude de la construction syntaxique d'entrées de dictionnaires terminologiques. Les segments extraits sont ensuite filtrés en fonction de leur fréquence dans le corpus afin d'éliminer les segments non répétés.

4.5 Confrontation avec les Notes de Communication Orale

Les textes auxquels nous nous intéressons dans cette thèse sont les Notes de Communication Orale qui sont des textes ayant des caractéristiques linguistiques particulières. Nous avons confronté les méthodes précédentes avec ces textes afin d'évaluer leur efficacité potentielle.

Les approches fondées sur une analyse syntaxique du texte pour extraire des syntagmes nominaux comme dans les méthodes linguistiques (TERMINO, LEXTER) et certaines méthodes hybrides (ACABIT, TERMS) ne sont pas adaptées aux Notes de Communication Orale en raison du manque d'efficacité des analyseurs lexicaux et syntaxiques sur ce type de textes (cf. section 2.3.1).

Les approches statistiques apparaissent mieux adaptées à des textes dont les phénomènes linguistiques échappent aux règles usuelles. En se fondant sur ce constat, il nous apparaît nécessaire de nous tourner vers des processus d'analyse statistique pour déterminer les candidats-termes de Notes de Communication Orale. Parmi les systèmes ayant recours à une approche statistique, nous avons finalement retenu le système ANA. Son idée fondatrice de pouvoir traiter tous les textes, qu'ils soient de bonne ou de mauvaise qualité lexicale ou syntaxique, situe *a priori* ce système comme une solution adaptée au traitement de Notes de Communication Orale. Cette intuition est confirmée en étudiant de près sa méthodologie : l'absence totale de traitements linguistiques (alors que c'est le cas dans certains systèmes à base statistique comme XTRACT), la notion d'apprentissage autonome (sans apport extérieur : corpus d'entraînement, fichiers d'amorces prédéfinis) le prédispose au traitement de tout type de textes sans distinction, et enfin le recours au principe de l'égalité souple lui permet de traiter correctement des textes contenant des fautes lexicales.

De par sa méthodologie fondée sur des principes statistiques, le système MANTEX pourrait également être adapté au traitement de Notes de Communication Orale. Nous avons néanmoins choisi d'utiliser ANA plutôt que MANTEX en raison de la disponibilité et de l'implication de sa conceptrice Chantal Enguehard tout au long de nos travaux de recherche.

Partie II

MeGET : une méthode d'extraction fondée sur un modèle de connaissances

Somewhere over the rainbow skies are blue, and the dreams that you dare to dream really do come true. Some day I'll wish upon a star and wake up where the clouds are far behind me, where troubles melt like lemondrops. Away above the chimney tops, that's where you'll find me.[1]
— Judy Garland, The Wizard of Oz.

[1] *Quelque part, au-delà de l'arc-en-ciel, les cieux sont bleus, et les rêves que vous osez rêver deviennent vraiment réalité. Un jour je ferai un souhait en regardant une étoile, et je me réveillerai là où les nuages seront loin derrière moi, là où les ennuis fondent tels des gouttes de citron. Bien au-dessus des cheminées, c'est là où vous me trouverez.*

Chapitre 5

Élaboration de l'ontologie des besoins

Présentation

« *Un modèle conceptuel constitue à la fois un cadre pour comprendre et interpréter les informations provenant des experts et un langage pour les formaliser en vue de construire un système* » [Aussenac-Gilles & al. 1992].

La détermination préalable des connaissances à traiter en fonction des objectifs de l'application est à la base de notre méthode de modélisation. L'ontologie d'extraction (cf. section 3.4), conçue pour servir de base de connaissance au processus d'extraction à proprement parler, est construite autour d'une première ontologie élaborée à partir des informations à rechercher. Cette première ontologie est appelée ***ontologie des besoins***. Sa construction est réalisée en exprimant les informations à extraire sous la forme de prédicats. Dans cette ontologie les concepts sont nommés et décrits formellement mais ne sont pas lexicalisés.

Dans ce chapitre nous discuterons d'abord de la meilleure façon d'exprimer les informations à rechercher (section 5.1). Ensuite, après avoir défini le formalisme utilisé pour décrire l'ontologie d'extraction (section 5.2), nous présentons la méthode de construction de l'ontologie des besoins réalisée en deux étapes : la formalisation de chacune des informations à rechercher par un prédicat menant à la définition d'un premier modèle dit *modèle prédicatif* (section 5.3) et l'extension de ce modèle par la description détaillée des concepts mis en jeu dans les prédicats (section 5.4). La figure 5.1 résume le processus d'élaboration de l'ontologie des besoins.

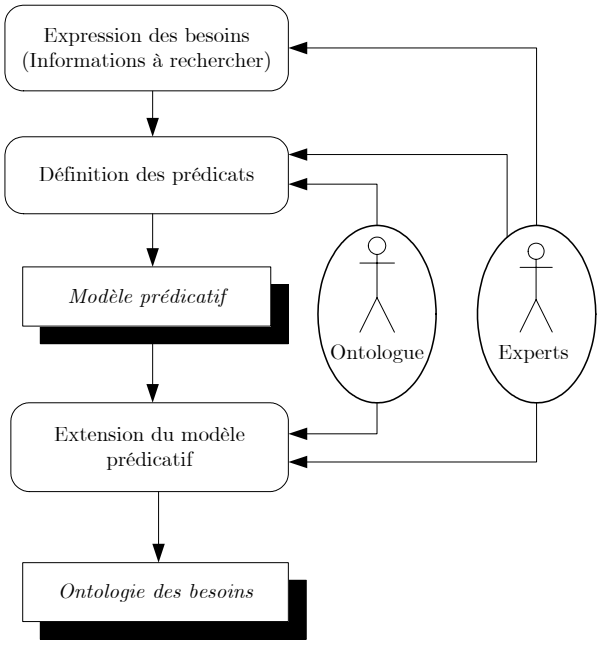

Figure 5.1 : Processus d'élaboration de l'ontologie des besoins

5.1 Expression des informations à rechercher

La construction de l'ontologie des besoins est réalisée conjointement avec des experts capables d'exprimer précisément les informations à rechercher dans les textes. Ces experts doivent également connaître les corpus à traiter ainsi que le domaine dans lequel ils s'inscrivent. En effet une certaine connaissance du contenu des textes est nécessaire pour ne pas envisager d'y chercher des éléments totalement hors de propos (comme des informations n'y apparaissant jamais).

Le travail d'expression des informations à rechercher doit être guidé par des techniques précises car les façons d'exprimer des connaissances peuvent différer très sensiblement d'un expert à un autre ou d'un domaine à un autre. Or le but de notre approche est de définir une méthodologie générique permettant de traiter des Notes de Communication Orale de différents domaines en s'appuyant sur une expertise humaine. Aussi nous devons spécifier un cadre formel dans lequel seront décrites les informations à rechercher et définis les concepts et les relations qu'elles expriment.

5.1.1 Des formulaires pour décrire les informations

Dans les systèmes usuels d'Extraction d'Information, les informations à rechercher sont définies de manière structurée grâce à des formulaires d'extraction dont les champs expriment les différents éléments de ces informations. Un formulaire formalise un type précis

d'information (par exemple une vente, un attentat, etc.). Il peut être défini de deux manières :

- Uniquement par des champs valués par des entités simples (entités nommées, dates, montant, etc.), on parlera alors de formulaire simple (cf. exemple 5.1) ;
- Par une combinaison de champs valués par des entités simples et de champs valués par des pointeurs sur d'autres formulaires (comme les *références* définies dans les formulaires de la campagne d'évaluation MUC-5, cf. section 1.3.2). On parlera dans ce cas de formulaire complexe (cf. exemple 5.2). Un formulaire de ce type s'apparente à une *Frame* [Gruber 1993] dans laquelle les valeurs des attributs peuvent être elles-mêmes des *Frames*.

Dans les exemples 5.1 et 5.2 **[NOM_FORMULAIRE]** définit un formulaire et « nom_entité » un élément désignant une catégorie d'entités simples, *champ* exprime le nom des champs à remplir par le système.

Exemple 5.1 (Formulaire simple)

[DIRIGEANT_ENTREPRISE]

Nom	:	« nom de personne »
Prénom	:	« prénom de personne »
Entreprise	:	« nom d'entreprise »
Grade	:	« catégorie de dirigeant »

Exemple 5.2 (Formulaire complexe)

[VENTE]

Id_Vente	:	« numéro »
Acheteur	:	**[INDIVIDU]**
Vendeur	:	**[INDIVIDU]**
Objet	:	**[PRODUIT]**

[INDIVIDU]

Id_Individu	:	« numéro »
Nom	:	« nom de personne »
Prénom	:	« prénom de personne »

[PRODUIT]

Id_Produit	:	« numéro »
Nom produit	:	« nom de produit »
Producteur	:	« nom d'entreprise »
Prix	:	« valeur monétaire »

Les formulaires présentent l'avantage d'exprimer formellement et précisément ce qui est recherché. Hors d'un tel cadre formel, une même information peut être exprimée de très nombreuses façons (différentes phrases en langue naturelle, ensemble de questions, liste d'éléments ou de termes, requête, etc.) et il serait difficile pour un système de traiter toutes ces formes[1]. De plus un formalisme structuré exprimant l'information à rechercher est à même de s'intégrer à une application informatique visant à extraire automatiquement des connaissances à partir de textes.

Les exemples les plus caractéristiques de formulaires d'extraction sont ceux utilisés dans les conférences MUC. Un exemple en anglais en est donné au chapitre 1 (cf. exemple 1.3, page 26). Ce type de formulaire est à la base de nombreuses méthodes d'Extraction d'Information des années 1990 car le respect de son format était impératif pour participer aux campagnes d'évaluation MUC. Le format des formulaires MUC est également repris dans des systèmes plus récents comme dans l'application FirstInvest développée par Thierry Poibeau [Poibeau 2002] et traitant un fil de presse financier (suite de dépêches) afin d'extraire des informations sur des opérations financières.

Dans l'application FirstInvest, les concepteurs ont d'abord défini un ensemble de formulaires simples mais après consultation des experts, il s'est avéré que de tels formulaires ne permettaient pas de représenter toute l'information recherchée. Les concepteurs se sont alors tournés vers des formulaires plus élaborés (formulaires complexes). Cet exemple illustre la difficulté d'élaborer des formulaires d'extraction [Wilks & al. 1997] et la nécessité de réaliser ce travail de manière conjointe avec un expert du domaine concerné par les textes, non seulement pour connaître les informations à rechercher mais également pour l'élaboration de leur représentation formelle.

5.1.2 Exprimer les informations à extraire par des prédicats

Dans le cadre de **MEGET**, nous avons choisi d'élaborer des patrons d'information se substituant aux formulaires traditionnellement utilisés dans les systèmes d'Extraction d'Information. Chaque patron correspond à un type d'information à rechercher et est élaboré en collaboration avec des experts possédant une grande connaissance des informations qui doivent être extraites. Il s'agit d'experts du domaine et, la plupart du temps, de professionnels des technologies de l'information mais pas de spécialistes en informatique ou en linguistique [Poibeau & Balvet 2001]. Aussi pour nous détacher d'éventuels problèmes de mauvaise compréhension dus aux différences culturelles ou conceptuelles entre le concepteur du système (connaissances informatiques, ontologiques et linguistiques) et l'expert (connaissances du domaine), nous choisissons de construire les patrons en suivant un formalisme prédéfini. Ce formalisme doit être à la fois simple pour être facilement appréhendable par les experts, et suffisamment élaboré pour rendre compte des informations à rechercher dans toute leur complexité.

[1] En Traitement Automatique de la Langue Naturelle ainsi que dans le domaine de l'Extraction de Connaissances, des travaux sont menés pour traiter chacune des différentes façons d'exprimer la recherche d'une ou plusieurs informations comme par exemple la Tâche de Question/Réponse pour la formulation par une question ou encore la Recherche d'Information traitant des requêtes (cf. section 1.2).

Nous représentons chaque patron d'information par un prédicat, c'est-à-dire une structure logique possédant un ensemble d'arguments. Un prédicat formalise donc un type précis d'information. Chaque argument est l'équivalent d'un champ de formulaire, et devra être valué par une information issue du texte. Le format de ces prédicats ainsi que la façon de les élaborer sont décrits en détail dans la section 5.3.

Le choix de recourir à des prédicats est motivé par leur lisibilité, leur capacité à exprimer des informations simples ou complexes et l'idée qu'il s'agit d'éléments adaptés à une opérationnalisation dans un cadre informatique. Un tel formalisme permet de guider concepteur et experts dans l'expression des besoins et d'obtenir facilement une représentation formelle de chacun d'eux.

Nous ne nous soucions pas de la valeur logique des prédicats (dépendante pour chacun d'eux de celle de ses arguments). En effet la notion de valeur de vérité n'a pas d'intérêt pour des arguments valués par des informations dans le cadre d'un processus d'Extraction d'Information : quand une information valuant un argument est trouvée lors de l'analyse d'un texte, il ne rentre pas dans les objectifs de l'Extraction d'Information de définir si cette information est vraie ou fausse. La détermination de la valeur de vérité d'une information présente dans un texte est un problème complexe et dépendant de nombreux critères (subjectivité du lecteur, nature et origine des textes, contexte historique ou politique des documents et du lecteur, etc.) qui dépasse largement le cadre de nos travaux.

L'expression des informations à rechercher est à la base de l'ontologie des besoins. Chaque prédicat représentant une information définit ainsi un concept de cette ontologie.

5.2 Formalisme de représentation de l'ontologie

Afin de rendre opérationnelle (cf. section 3.2.4.1) l'ontologie modélisant les informations à rechercher, nous avons opté pour une représentation sous la forme d'une grammaire formelle dont l'ensemble des règles (règles formelles) constitueront la base de connaissance du système d'extraction. Chacune des règles formelles décrit un concept de l'ontologie. La définition d'un concept est donnée par une expression définissant ce qui le compose, ce qui le caractérise et ses relations avec d'autres concepts du modèle.

Nous nous sommes inspiré de la notation BNF (Backus Naur Form aussi appelée Backus Normal Form)[2] pour spécifier les notations de notre formalisme de représentation.

[2] Cette notation a été établie originellement par John Backus et Peter Naur pour définir le langage de programmation Algol 60 [Naur & al. 1960] et a connu depuis plusieurs extensions aboutissant à la définition d'une norme BNF étendue (Extended BNF ou EBNF) [Scowen 1996]. La notation BNF est une des notations méta-syntaxiques les plus couramment utilisées pour décrire la syntaxe des langages de programmation.

Dans notre formalisme de représentation, une grammaire est définie comme un quadruplet (E_N, E_T, Ψ, K) où :

- E_N est l'ensemble des symboles non-terminaux. Les non-terminaux correspondent aux concepts de l'ontologie. Un non-terminal est formé par l'identifiant du concept auquel il correspond encadré par les signes « < » et « > » ;
- E_T l'ensemble des symboles terminaux. Les symboles terminaux correspondent à des unités lexicales présentes dans les textes (mots, termes, expressions numériques ou alphanumériques).

  L'intersection des symboles terminaux et non-terminaux est vide ($E_N \cap E_T = \varnothing$) ;
- Ψ l'ensemble des unités lexicales intervenant dans le formalisme et n'appartenant pas à $E_N \cup E_T$ (opérateurs, séparateurs, etc.) ;
- K l'ensemble des règles α ::= { β } ;; avec $\alpha \in E_N$ et β une expression composée d'éléments de $E_N \cup \Psi$ ou de $E_T \cup \Psi$.

  Dans une règle, le symbole « ::= » signifie « *est défini par* », des accolades délimitent l'expression située en partie droite et le double point-virgule indique la fin de la règle.

  Une règle α ::= { β } ;; où α est égal à <C> (avec C l'identifiant d'un concept) signifie que le concept C est défini par l'expression β qui exprime les éléments caractérisant C ainsi que la nature de ses relations avec d'autres concepts de l'ontologie.

  La partie droite d'une règle ne peut contenir à la fois des non-terminaux et des terminaux, c'est-à-dire qu'elle ne peut mêler des concepts avec des éléments des textes. Cette restriction est fixée pour séparer les concepts de leurs représentations lexicales, c'est-à-dire différencier dans la représentation de l'ontologie le niveau conceptuel du niveau linguistique et faciliter ainsi son opérationnalisation.

Certains concepts de l'ontologie sont issus des prédicats exprimant les informations recherchées, d'autres apparaissent comme nécessaires pour modéliser le domaine dans lequel s'inscrivent les informations recherchées. Les concepts sont donc définis par deux types de règles formelles : les règles prédicatives et les règles constitutives. Ces règles permettent de décrire la totalité du modèle et sont présentées plus avant lors de la description de la construction de l'ontologie.

5.3 Définition des prédicats

La description conceptuelle des informations à extraire est réalisée grâce à un ensemble de prédicats. Cette première étape de la construction de l'ontologie des besoins est appelée étape de ***définition des prédicats*** et est le pendant dans **MEGET** de la phase de définition de formulaires d'extraction d'un système classique d'Extraction d'Information.

Chacune des informations à extraire est formalisée par un prédicat. Chaque prédicat définit un unique concept de l'ontologie. En se reportant aux définitions énoncées dans le chapitre 3 (cf. section 3.2.2.1), un prédicat P définissant un concept C correspond à une représentation de l'intension de C, c'est-à-dire l'ensemble des éléments caractérisant ce concept. Corollairement toute instance de P définit une unique instance du concept C.

Chacun des concepts C définis par un prédicat est décrit par une règle <C> ::= { β } ;; où l'expression β est un prédicat. Les règles de ce type sont appelés règles prédicatives.

Dans cette section, nous présentons les notations conceptuelles utilisées dans **MEGET**, les différents éléments composant un prédicat ainsi que les règles prédicatives. Les exemples donnés sont issus d'un travail d'expérimentation effectué sur le corpus [CREC], un corpus de Notes de Communication Orale en langue française issu d'une banque (cf. chapitre 8).

5.3.1 Notations

L'identifiant d'un concept ou d'une relation joue non seulement sur la compréhension de l'ontologie par l'utilisateur mais aussi sur celle du cogniticien en cours de modélisation ou lors de la maintenance de l'ontologie en cas de correction ou de modification du modèle [Aussenac-Gilles & al. 2000a,b]. Les identifiants nommant les concepts et les relations lors de la modélisation doivent expliciter clairement leur signification. Ce principe s'applique aux concepts définis par des prédicats et plus généralement à tous les concepts du modèle.

Dans **MEGET**, l'identifiant d'un concept Y, décrivant une notion N, est formé du terme vedette de Y écrit en majuscule et préfixé par « C_ » . Par exemple le concept décrivant la notion d'*achat* aura comme identifiant C_ACHAT (*achat* étant le terme le plus usité pour lexicaliser cette notion).

De plus pour simplifier les notations, lorsqu'un concept C_Y est défini par un prédicat, nous nommerons ce prédicat P_Y. En reprenant l'exemple précédent, le prédicat définissant le concept C_ACHAT sera nommé P_ACHAT.

De même, une relation R entre plusieurs concepts aura comme identifiant un ou plusieurs termes de sa terminologie séparés par le symbole « _ » et préfixés par « R_ » (*R_est_date_de* par exemple).

5.3.2 Format des prédicats

Dans **MEGET**, un prédicat est une structure repérée par un identifiant et formée de plusieurs arguments : un descripteur, un objet et un ensemble d'options.

5.3.2.1 Descripteur du prédicat

Le ***descripteur*** d'un prédicat est un élément nécessaire à la description du concept défini par ce prédicat. Un concept défini par un prédicat élaboré sans descripteur ou un prédicat avec une valeur nulle pour cet argument, ne peut exister dans une ontologie

construite avec **MEGET** : le descripteur est un argument obligatoire lors de la définition d'un prédicat.

La valeur du descripteur d'un prédicat P_Y est une représentation conceptuelle des unités lexicales permettant de lexicaliser le concept C_Y, c'est-à-dire les éléments de la terminologie de C_Y. Il s'agit soit d'un unique concept qui conceptualise les lexicalisations de C_Y dans les textes, soit du père d'une hiérarchie de concepts dont les derniers fils conceptualisent des éléments de la terminologie de C_Y. Par exemple un prédicat décrivant le concept C_ACHAT aura comme descripteur un concept C_DESCRIPTEUR_ACHAT qui conceptualise des termes comme "*achat* ", "*acheter* ", "*rachat* ", etc. Le prédicat P_ACHAT peut alors s'écrire[3] :

```
P_ACHAT (descripteur = C_DESCRIPTEUR_ACHAT)
```

Lors de la construction de l'ontologie des besoins, nous travaillons sans nous intéresser au contenu des textes, c'est-à-dire sans nous soucier des mots ou termes correspondant aux concepts. Aussi lors de cette étape les concepts utilisés comme valeur de descripteurs sont définis sans que soient déterminées les unités lexicales qu'ils conceptualisent. Ils sont définis uniquement par un identifiant et une intension exprimée en langue naturelle. Les unités lexicales sont déterminées ultérieurement lors de l'étude termino-ontologique (cf. section 6.2.3.2).

Cette méthode possède l'avantage de ne pas lier la définition des prédicats à un corpus précis en ne fixant pas *a priori* les lexicalisations des concepts car celles-ci sont susceptibles de varier d'un corpus à l'autre. C'est particulièrement vrai dans les Notes de Communication Orale où il existe des particularités linguistiques propres à chaque corpus voire même à chaque texte. Ainsi dans un ensemble de Notes de Communication Orale, un même trait sémantique peut être représenté par des lexèmes spécifiques à certains textes et pas à d'autres comme les abréviations dont la forme est dépendante des rédacteurs ou par de nombreuses variantes lexicales d'un mot ou terme désignant ce trait sémantique (variantes quelquefois issues de fautes orthographiques). Par exemple dans notre expérimentation, les lexicalisations du concept C_ACHAT sont différentes d'un texte à un autre : dans un texte il s'agira des mots "*achat* ", "*acheter* " ou de l'abréviation "*acht* ", dans un autre des mots "*rachat* ", "*acquisition* ", "*acquérir* " ou "*aquerir* " et dans un troisième de l'abréviation "*acqtion* ". Ainsi les éléments conceptualisés par le concept C_DESCRIPTEUR_ACHAT dépendront des textes traités.

La valeur du descripteur d'un prédicat P_Y est un unique concept C_D_Y qui ne peut être la valeur d'aucun autre argument de prédicat (de P_Y ou d'un autre prédicat). En effet, un tel concept C_D_Y est défini exclusivement pour valuer le descripteur de P_Y et n'a pas d'autre fonction dans l'ontologie.

Notons que cette restriction interdit qu'un même concept C_D_Y value le descripteur de plusieurs prédicats. En effet, pour éviter des ambiguïtés sémantiques, il n'est pas souhaitable qu'une unité lexicale conceptualisée par C_D_Y puisse lexicaliser plusieurs concepts différents.

[3] Pour simplifier la lecture, nous présenterons les prédicats avec une syntaxe inspirée de Prolog.

5.3.2.2 Objet du prédicat

L'***objet*** d'un prédicat P_Y décrit l'ensemble des concepts C_X_i sur lesquels peut porter la notion exprimée par C_Y [4]. Chaque concept C_X_i est appelé « *type d'objet* » de P_Y. L'ensemble des types d'objet de P_Y est nommé T_Objet(P_Y).

En utilisant une analogie avec la grammaire française, l'argument objet pour un prédicat est similaire au Complément d'Objet Direct (COD) pour un verbe.

La valeur de l'objet d'une instance d'un prédicat P_Y est une instance d'un des concepts de T_Objet(P_Y).

L'objet décrit une relation *R_est_objet_de* entre chaque concept C_X_i, type d'objet d'un prédicat P_Y, et C_Y. L'intension de cette relation est définie comme « *lien entre une notion et un des éléments sur laquelle elle porte* ». Cette relation apporte une information sur la description de C_Y. L'argument objet de P_Y correspond donc à un attribut [Gomez-Pérez 2004] particulier du concept C_Y (attribut objet de C_Y).

L'objet n'est pas nécessaire à la définition d'un prédicat mais, lorsqu'il est présent, il doit être instancié obligatoirement. La valeur de l'objet d'une instance de prédicat ne peut pas être nulle.

L'exemple 5.3 illustre la notion d'objet en reprenant le prédicat P_ACHAT.

Exemple 5.3 (Objet du prédicat P_ACHAT)

Les éléments sur lesquels peuvent porter un achat sont des véhicules motorisés ou non (voiture, moto, vélo, bateau, etc.), des biens immobiliers (maison, appartement, etc.) et des produits émis par des organismes financiers (prêts, assurances, etc.). Ils sont respectivement conceptualisés par les concepts C_VEHICULE, C_IMMOBILIER et C_PRODUIT_BANCAIRE. Ainsi : T_Objet(P_ACHAT) = {C_VEHICULE, C_IMMOBILIER, C_PRODUIT_BANCAIRE}.

Le prédicat P_ACHAT peut alors s'écrire :

```
P_ACHAT ( descripteur =  C_DESCRIPTEUR_ACHAT ,
          objet  =  {C_VEHICULE, C_IMMOBILIER,
                     C_PRODUIT_BANCAIRE}
          )
```

La figure 5.2 illustre les relations décrites par l'objet du prédicat P_ACHAT.

[4] À l'image du *focus* dans les systèmes de Question-Réponse (cf. section 1.2.4.1.1).

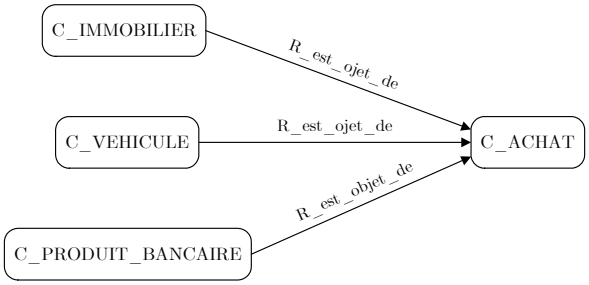

Figure 5.2 : Illustration de la définition de l'objet du prédicat P_ACHAT

À la différence du descripteur, un concept C_OBJ, type d'objet d'un prédicat P_Y peut également être un type d'objet d'autres prédicats (cf. exemple 5.4) ou bien être valeur d'une option de prédicat (cf. section 5.3.2.3).

Exemple 5.4 (Concept type d'objet de plusieurs prédicats)

Le concept C_VEHICULE est un type d'objet du prédicat P_ACHAT, mais il est également un type d'objet du prédicat P_VENTE ainsi que du prédicat P_VOL. Ces deux prédicats définissent respectivement les concepts C_VENTE exprimant la notion de « vente de produit manufacturé ou financier », et C_VOL exprimant la notion de « vol de bien matériel ». Il existe ainsi une relation R_est_objet_de entre le concept C_VEHICULE et chacun de ces concepts (cf. figure 5.3).

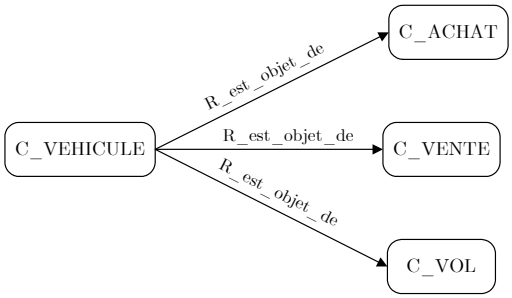

Figure 5.3 : Concepts définis par des prédicats ayant le concept C_VEHICULE comme type d'objet

À ce point de la phase de modélisation, les concepts déterminés comme type d'objet de prédicat sont définis avec les experts de manière informelle (à l'exception de ceux définis par un prédicat). Chacun des concepts est nommé par un identifiant et son intension est exprimée en langue naturelle (par exemple « *un C_VEHICULE est un véhicule motorisé* »). La description formelle des concepts est réalisée lors de la phase d'extension du modèle prédicatif (cf. section 5.4).

L'exemple 5.5 illustre la notion d'objet et les relations inter-concepts qu'elle exprime pour le concept C_PROJET défini lors de notre expérimentation. Nous constatons ici que plusieurs concepts définis par un prédicat sont types d'objet d'autres prédicats (à l'image des formulaires complexes d'extraction dans lesquels les champs peuvent être valués par d'autres formulaires).

Exemple 5.5 (Objet du prédicat P_PROJET)

Le concept C_PROJET est défini par le prédicat P_PROJET et modélise la notion de projet d'un client, c'est-à-dire tous les projets éventuels d'un client. Il peut s'agir de projet d'achat ou de vente de biens (matériels, financiers, immobiliers), de projet immobilier (projet de construction immobilière ou projet concernant un bien immobilier mais pas encore défini par le client[5]), d'un projet non défini concernant un véhicule[6] ou encore d'un projet d'action financière (ouverture d'un livret, d'un compte, achat d'actions). Le travail avec les experts nous a amené à définir les types d'objet du prédicat P_PROJET par l'ensemble suivant :

```
T_Objet(P_PROJET) = {C_ACHAT, C_VENTE, C_IMMOBILIER, C_VEHICULE,
                     C_ACTION_BANCAIRE}
```

avec C_ACHAT et C_VENTE des concepts définis par un prédicat.

La figure 5.4 présente les relations entre concepts décrites par l'argument objet du prédicat P_PROJET.

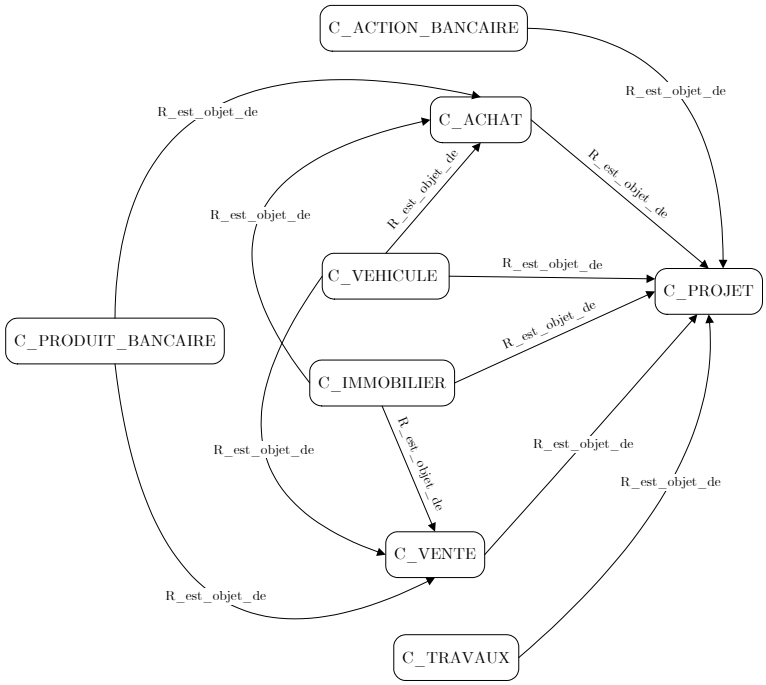

Figure 5.4 : Illustration des relations décrites par l'objet du prédicat P_PROJET

[5] Un cas typique est celui d'un client désirant changer de logement mais ne sachant pas encore s'il achètera ou louera un logement neuf ou ancien ou s'il fera construire.

[6] Par exemple un client hésitant entre l'achat d'un deuxième véhicule ou le remplacement du premier (achat d'un nouveau véhicule avec éventuellement vente du premier).

5.3.2.3 Arguments optionnels

La définition d'un prédicat peut s'accompagner d'un ou plusieurs arguments optionnels dits ***options***. Ces arguments complètent le prédicat mais ne sont pas nécessaires ni suffisants à sa définition. Leur nombre varie d'un prédicat à l'autre.

Chaque option opY_i d'un prédicat P_Y exprime une relation informative R_opY_i entre le concept C_Y et chaque élément C_O_i d'un ensemble de concepts de l'ontologie {C_O_1, C_O_2, …, C_O_n}. Une relation informative apporte à la description d'un concept une information ni nécessaire ni suffisante. Chaque concept C_O_i est appelé « *type de l'option* opY_i » de P_Y. L'ensemble des types de l'option opY_i de P_Y est nommé $T\_opY_i(P\_Y)$.

Chaque argument optionnel d'un prédicat P_Y correspond à un attribut du concept C_Y. Par exemple une option *datation* d'un prédicat P_ACHAT exprimant une relation *R_est_date_de*, entre C_ACHAT et des concepts du modèle correspond à un attribut du concept C_ACHAT renseignant sur la date de l'achat.

En reprenant l'analogie avec la grammaire française évoquée pour l'objet (cf. section 5.3.2.2), les options pour un prédicat sont comparables aux Compléments Circonstanciels (de Lieu, de Temps, Moyen, But, etc.) pour un verbe.

Au niveau des instances, une option d'une instance de P_Y sera valuée par une instance d'un des types de cette option. Au contraire de l'argument objet, une instance de prédicat peut exister avec une ou plusieurs options possédant une valeur nulle.

Le nom d'une option opY_i est choisi afin d'exprimer explicitement la nature de la relation R_opY_i traduite par opY_i. Il s'agit généralement d'un terme issu de l'identifiant de la relation (par exemple la relation *R_est_localisation_de* est traduite par l'option « *localisation* »).

Comme pour l'objet, les concepts déterminés comme type de chacune des options d'un prédicat sont définis ici de manière informelle par un identifiant et une intension exprimée en langue naturelle. Les définitions formelles de ces concepts interviendront dans la phase d'extension du modèle prédicatif.

L'exemple 5.6 illustre la notion d'argument optionnel en prenant comme exemple le prédicat P_ACHAT.

Exemple 5.6 (Options du prédicat P_ACHAT)

Le concept C_ACHAT est défini par le prédicat P_ACHAT. Ce prédicat est élaboré avec trois options renseignant sur la date de l'achat, le montant du produit acheté et la localisation du produit. Ces options sont appelées respectivement datation, montant et localisation. Les types de chacune de ces options sont : le concept C_DATE (exprimant une date) pour la datation, le concept C_SOMME (exprimant une somme monétaire) pour le montant et le concept C_LIEU (décrivant un lieu géographique) pour la localisation.

Le prédicat P_ACHAT peut alors s'écrire :

```
P_ACHAT (descripteur    =  C_DESCRIPTEUR_ACHAT,
         objet  =  {C_VEHICULE, C_IMMOBILIER,C_PRODUIT_BANCAIRE},
         datation = {C_DATE},
         montant =  {C_SOMME},
         localisation  =  {C_LIEU} )
```

Il existe une relation R_est_la_date_de entre le concept C_DATE et le concept C_ACHAT, une relation R_est_le_montant_de entre les concepts C_SOMME et C_ACHAT et une relation R_est_la_localisation_de entre les concepts C_LIEU et C_ACHAT.

La figure 5.5 illustre graphiquement cet exemple.

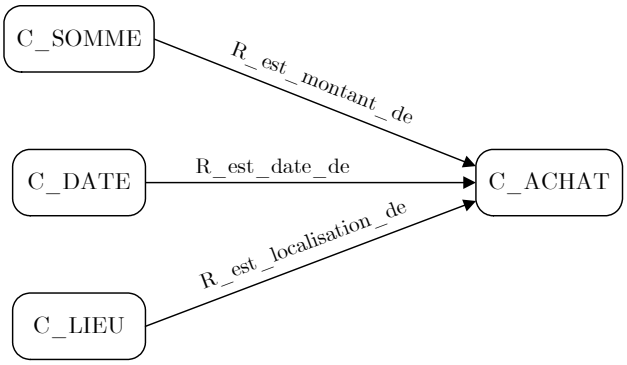

Figure 5.5 : Illustration des relations décrites par les options du prédicat P_ACHAT

5.3.2.4 Options identiques et différentes

Lorsqu'un concept C_B défini par un prédicat P_B est un type d'objet d'un prédicat P_A, le concepteur du modèle peut choisir qu'une option opA_i de P_A soit valuée par une information issue de P_B, c'est-à-dire par la valeur d'une option opB_j de P_B. Dans ce cas, les options opA_i et opB_j sont considérées « *identiques* » ($opA_i == opB_j$).

Un tel choix signifie que l'option opA_i d'une instance I_{P_A} de P_A aura pour valeur celle de l'option opB_j d'une instance I_{P_B} de P_B lorsque I_{P_A} a pour objet l'instance de C_B définie par I_{P_B}.

Pour assurer la cohérence et la lisibilité du modèle, le concepteur doit respecter les conditions suivantes lorsqu'il considère deux options opA_i et opB_j comme identiques :

- Le même nom sera donné à opA_i et à opB_j ;
- $T\_opB_j(P\_B) \subseteq T\_opA_i(P\_A)$
  Si un concept C_O de $T\_opB_j(P\_B)$ n'appartient pas à $T\_opA_i(P\_A)$, opA_i et opB_j ne peuvent être identiques car dans le cas contraire, il existe un risque de voir une instance de C_O valuer l'option opA_i, ce qui n'est pas cohérent avec le modèle.

Ces conditions sont nécessaires lorsque deux options sont identiques mais elles ne sont pas suffisantes : deux options opA_i et opB_j ayant le même nom et telles que $T\_opB_j(P\_B) \subseteq T\_opA_i(P\_A)$ ne sont pas nécessairement identiques. L'identité entre deux options résulte d'un choix du concepteur fait en fonction des objectifs de l'ontologie.

Exemple 5.7 (Options identiques)

Le prédicat P_PROJET est élaboré avec une option *datation* exprimant la date du projet avec T_datation(P_PROJET) = {C_DATE}. Si les concepteurs de l'ontologie considèrent la date d'un projet comme égale à la date d'un achat lorsque cet achat est l'objet du projet (c'est-à-dire lorsqu'il s'agit d'un projet d'achat), alors l'option *datation* du prédicat P_PROJET sera considérée identique à l'option *datation* du prédicat P_ACHAT.

Lorsque deux options opA_i de P_A et opB_j de P_B ne sont pas identiques, elles sont « *différentes* » ($opA_i \neq opB_j$). Lorsque $opA_i \neq opB_j$: il n'existe aucun lien entre la valeur de opA_i et celle de opB_j. Notons que deux options jugées différentes peuvent avoir le même nom.

Exemple 5.8 (Options différentes)

L'option datation du prédicat P_PROJET est par exemple considérée comme différente de l'option *datation* du prédicat P_REFUS (avec C_REFUS un concept exprimant la notion de refus de la part d'une banque à un client et C_PROJET un type d'objet de P_REFUS) car la date d'un projet n'est pas jugée comme la même que celle de son refus.

5.3.3 Expression formelle des prédicats

Les concepts définis par des prédicats sont décrits par des règles formelles appelées *règles prédicatives*. Chaque concept C_Y défini par un prédicat P_Y est décrit par une règle prédicative <C_Y> ::= {EXP[P_Y]} dans laquelle EXP[P_Y] est une expression formalisant le prédicat P_Y (cf. exemple 5.9). Une telle règle se présente selon la forme décrite par la figure 5.6.

<C_Y> ::= {

descripteur	=	**<C_D>**	;
objet	=	<C_X_1> | <C_X_2> | ... | <C_X_m>	;
opt_1	=	<C_$O1_1$> | <C_$O1_2$> | ... | <C_$O1_{n1}$>	;
opt_2	=	<C_$O2_1$> | <C_$O2_2$> | ... | <C_$O2_{n2}$>	;
...			;
opt_q	=	<C_Oq_1> | <C_Oq_2> | ... | <C_Oq_{nq}>	.

} ;;

avec « | » la marque de la disjonction : arg = C_A | C_B signifie que la valeur de arg est soit C_A soit C_B mais pas les deux (le symbole « | » est équivalent au OU exclusif)

Figure 5.6 : Règle prédicative

La règle de la figure 5.6 signifie que :

- Le concept C_D est la valeur du descripteur du prédicat P_Y ;
- T_Objet(P_Y) = {C_X_1, C_X_2, ..., C_X_m} avec $m \in \mathbb{N}^*$;
- Il existe q options *opt_k* décrivant chacune une relation entre C_Y et chaque concept C_Ok_i .

Un point-virgule sépare les définitions des différents arguments. Un point est placé après la définition du dernier argument et indique qu'il n'y en a plus d'autres.

Seul l'argument descripteur est présent quelle que soit la règle. La présence des autres arguments (objet et options) dépend de la définition du concept décrit par la règle.

Exemple 5.9 (Règle prédicative décrivant le concept C_ACHAT)

Le concept C_ACHAT est défini par le prédicat P_ACHAT de l'exemple 5.6. Ce prédicat est formalisé en suivant la syntaxe de notre formalisme de représentation par l'expression suivante :

descripteur = <C_DESCRIPTEUR_ACHAT> ;
objet = <C_VEHICULE> / <C_IMMOBILIER>
/ <C_PRODUIT BANCAIRE> ;
datation = <C_DATE> ;
montant= <C_SOMME> ;
localisation = <C_LIEU>.

Le concept C_ACHAT est alors décrit par la règle prédicative suivante :

```
<C_ACHAT> ::= {
                descripteur   = <C_DESCRIPTEUR_ACHAT> ;
                objet         = <C_VEHICULE> / <C_IMMOBILIER>
                                /<C_PRODUIT BANCAIRE> ;
                datation      = <C_DATE> ;
                montant       = <C_SOMME> ;
                localisation  = <C_LIEU>.
              } ;;
```

Lorsqu'une option opY d'un prédicat P_Y est considérée identique à une option $opOBJ_i$ d'un prédicat définissant un concept C_OBJ$_i$ ∈ T_Objet(P_Y), cette identité est représentée dans la règle par la présence du symbole « == » entre le nom de l'option opY et la liste des éléments de T_opA$_i$(P_Y) (cf. exemple 5.10). De plus les options opY et opX_i porteront le même nom dans les règles décrivant C_Y et C_OBJ$_i$.

Ce formalisme implique que chaque option $opOBJ_j$, définie pour un concept C_OBJ$_j$ appartenant à T_Objet(P_Y) et ayant le même nom que opY, est identique à opY. Aussi dans les cas où le concepteur considère $opOBJ_j$ différente de opY, le nom de $opOBJ_j$ devra être changé dans la règle décrivant C_OBJ$_j$.

Toutes les autres options sont écrites avec le symbole « = » entre leur nom et l'ensemble de leurs types (cf. exemple 5.10).

Exemple 5.10 (Règle prédicative avec options identiques)

```
<C_PROJET>  ::= {
                  descripteur =  <C_DESCRIPTEUR_PROJET> ;
                  objet       =  <C_ACHAT> / <C_IMMOBILIER>
                                 / <C_VENTE> / <C_VEHICULE>
                                 / <C_ACTION_BANCAIRE>
                                 / <C_TRAVAUX> ;
                  datation    == <C_DATE> ;
                  montant     =  <C_SOMME> ;
                  localisation =  <C_LIEU> .
                } ;;
```

Ici la date d'un projet est considérée égale à celle de l'achat lorsque le projet porte sur un achat ou à celle de la vente lorsqu'il s'agit d'un projet de vente (les autres concepts objets n'ont pas d'option date).
Par contre le montant et la localisation d'un projet sont considérées différents de ceux d'un achat ou d'une vente.

5.4 Extension du modèle prédicatif

L'extension du modèle prédicatif est la phase dans laquelle sont spécifiés les concepts valuant les arguments de chaque prédicat. En dehors de concepts génériques spécifiés à part (cf. section 5.4.2), ces concepts sont décrits par spécialisation ou par la définition de relations d'association avec d'autres concepts du modèle.

5.4.1 Description hiérarchique et associative des concepts

En nous appuyant sur un dialogue avec le ou les experts, nous décrivons les concepts issus du modèle prédicatif en définissant leurs relations avec d'autres concepts, ces nouveaux concepts étant eux-mêmes décrits de la même façon (description récursive des concepts). Les relations permettant de décrire les concepts sont de deux types : la relation de subsomption et la relation d'association. Une relation d'association entre deux concepts A et B signifie que l'un (A) a besoin de l'autre (B) pour se définir.

Lors de cette phase, nous cherchons également à définir les relations informatives (relations qui complètent la description d'un concept mais qui ne sont ni nécessaires ni suffisantes à sa définition) qui peuvent exister entre différents concepts de l'ontologie des besoins.

Notons que nous ne cherchons pas, à ce point de la modélisation, à lier les concepts à des entités textuelles c'est-à-dire à déterminer leurs représentations lexicales. Ce processus sera réalisé lors de la phase d'unification.

Chaque concept est décrit par une unique règle formelle qui traduit ses relations avec d'autres concepts de l'ontologie. Une telle règle est appelée règle constitutive. Dans une règle constitutive, un concept C_Y est défini par une expression β formée d'un ensemble de n concepts liés par un opérateur (<C_Y> ::= { β } ;;). L'ensemble des concepts contenus dans β est appelé *ensemble de définition* de la règle décrivant C_Y. Notre formalisme de représentation intègre trois types de règles constitutives : les règles sélectives, les règles conjonctives et les règles disjonctives [Even & Enguehard 2002].

5.4.1.1 Règles sélectives

Les règles sélectives sont du type :

$$<C\_Y> ::= \{ <C\_A_1> | < C\_A_2> | \ldots | < C\_A_n> \};; \quad \text{avec } n \in \mathbb{N}^+$$

Comme dans les règles prédicatives, l'opérateur « | » est équivalent au OU exclusif et exprime ainsi la notion de disjonction exclusive : la valeur de l'expression $C\_A_1 | C\_A_2$ est soit C_A_1, soit C_A_2 mais pas les deux. La règle encadrée ci-dessus signifie donc que C_Y est défini par la présence de l'un des concepts C_A_i.

Une règle sélective décrivant un concept C_Y exprime une relation de subsomption entre C_Y et chacun des concepts C_A_i de son ensemble de définition : C_Y subsume chaque concept C_A_i. La règle précédente signifie donc « C_A_1 est une sorte de C_Y », « C_A_2 est une sorte de C_Y », etc.

Un concept appartenant à l'ensemble de définition d'une règle sélective ne peut définir qu'un seul et unique concept : il ne peut donc pas appartenir à l'ensemble de définition d'une autre règle sélective. Cette contrainte vise à éviter les ambiguïtés lors de l'identification des instances de concepts dans les textes.

Exemple 5.11 (Règle sélective)

<C_VEHICULE> ::= { <C_AUTO> / <C_MOTO> / <C_VEHICULE_LOURD>} ;;

Le concept C_VEHICULE subsume les concepts C_AUTO, C_MOTO et C_VEHICULE_LOURD.

Un concept particulier dit *descripteur du concept C_Y* peut être utilisé lorsqu'un concept C_Y est identifiable dans les textes par une de ses productions lexicales mais également par un des éléments de sa terminologie, et que le concepteur désire séparer ces deux ensembles dans le modèle (par exemple séparer les termes "*maison*", "*t1*", "*appart*" qui sont des productions lexicales du concept C_IMMOBILIER, des termes "*immo*" ou "*immobilier*" qui sont des éléments de sa terminologie). Les productions lexicales seront conceptualisées par des concepts liés à C_Y, et la terminologie par le descripteur. À ce point de la modélisation, le descripteur est uniquement exprimé par un identifiant composé de Y préfixé par « C_DC_ ».

Un concept C_Y peut être ainsi être décrit par une règle sélective dans laquelle est présente un descripteur C_DC_Y conceptualisant la terminologie de C_Y (comme dans l'exemple 5.12).

Exemple 5.12 (Règle sélective avec un descripteur de concept)

<C_IMMOBILIER> ::= {
<C_DC_IMMOBILIER>
/ <C_APPARTEMENT>
/ <C_MAISON>
/ <C_DIVERS_IMMOBILIER>
} ;;

où C_DC_IMMOBILIER est un descripteur de concept. Il conceptualise la terminologie de C_IMMOBILIER ("immo", "immobilier", etc.).

5.4.1.2 Règles conjonctives

Les règles conjonctives sont du type

$$<C\_Y> ::= \{ <C\_A_1> + <C\_A_2> + \ldots + <C\_A_n> \} ;; \quad \text{avec } n \in \mathbb{N}^+$$

L'opérateur « + » correspond à la conjonction classique : la valeur de l'expression $C\_A_1 + C\_A_2$ est « C_A_1 ET C_A_2 ».

Une règle conjonctive traduit une relation d'association entre le concept C_Y qu'elle décrit et la totalité de son ensemble de définition {C_A_1, C_A_2, …, C_A_n}, c'est-à-dire qu'il existe une relation d'association entre C_Y et chacun des concepts C_A_i. Une telle relation signifie que C_Y a nécessairement besoin de chaque C_A_i pour être défini (cette relation se rapproche de la notion d'agrégation entre un composé et ses composants).

Par conséquent une instance de C_Y est définie par la présence d'une instance de tous les concepts C_A_i.

Exemple 5.13 (Règle conjonctive)

<C_PRET_VEHICULE> ::= {<C_DC_PRET> + <C_VEHICULE>} ;;

Un C_DC_PRET (descripteur de prêt) et un C_VEHICULE sont tous les deux nécessaires à l'existence d'un C_PRET_VEHICULE (conceptualisant la notion de « prêt d'argent pour un véhicule motorisé »).

5.4.1.3 Règles disjonctives

Les règles disjonctives sont du type

$$<C\_Y> ::= \{ <C\_A_1> \vee < C\_A_2> \vee \ \ldots \vee \ < C\_A_n> \} ;; \quad \text{avec } n \in \mathbb{N}^+$$

L'opérateur « $\vee$ » correspond à la disjonction classique : la valeur de l'expression $C\_A_1 \vee C\_A_2$ est soit C_A_1, soit C_A_2, soit les deux (C_A_1 ET C_A_2).

Une règle disjonctive traduit une relation d'association entre le concept C_Y qu'elle décrit et une sous-partie Δ composée d'un ou de plusieurs concepts C_A_i de son ensemble de définition {C_A_1, C_A_2, …, C_A_n}, c'est-à-dire qu'il existe une relation d'association entre C_Y et chacun des concepts C_A_j de Δ. Ainsi une règle disjonctive décrivant un concept C_Y signifie que la présence de n'importe quelle sous-partie de son ensemble de définition est suffisante pour définir C_Y.

Une instance de C_Y est donc soit une instance d'un concept C_A_i, soit une combinaison d'instances de plusieurs concepts C_A_i.

Une règle disjonctive permet de décrire des concepts qui regroupent plusieurs informations sans pour autant qu'il soit nécessaire de disposer de toutes ces informations pour les définir (cf. exemple 5.14).

La présence d'un seul concept C_A_i appartenant à l'ensemble de définition d'une règle disjonctive est suffisante pour définir le concept C_Y décrit par la règle. Aussi, pour éviter les ambiguïtés (afin qu'une instance de C_A_i ne puisse pas définir à la fois une instance de C_Y et d'un autre concept), C_A_i ne pourra appartenir à l'ensemble de définition d'une autre règle disjonctive ou d'une règle sélective. Réciproquement, un concept appartenant à l'ensemble de définition d'une règle sélective ne pourra être présent dans celui d'une règle disjonctive.

Ces restrictions ne concernent pas les règles conjonctives car la présence d'un seul concept de l'ensemble de définition d'une règle conjonctive n'est pas suffisante pour définir le concept décrit par cette règle.

Exemple 5.14 (Concept C_PERSONNE)

Une instance du concept C_PERSONNE correspond à un individu. Un individu peut être décrit en fonction du contexte par une instance du concept C_PRENOM (conceptualisant la notion de « prénom »), par une instance du concept C_NOM (conceptualisant la notion de « nom de famille ») mais également par une instance de la combinaison de ces deux concepts. Ainsi le concept C_PERSONNE est décrit par la règle disjonctive suivante :

<C_PERSONNE> ::= { <C_NOM> ∨ <C_PRENOM> } ;;

5.4.1.4 Définition de relations informatives

Il est possible de définir des relations informatives entre les concepts décrits par des règles constitutives et d'autres concepts du domaine, à l'instar de celles qui sont définies par les arguments optionnels des prédicats (cf. exemple 5.15). Ces relations apportent des informations sur un concept et complètent sa description mais ne sont ni nécessaires ni suffisantes à la définition de ce concept. Le type d'information exprimé par une telle relation pour un concept C_Y correspond à un attribut de ce concept. Par exemple la relation *R_est_couleur_de*, entre un concept C_COLORI (conceptualisant un ensemble de couleurs) et un concept C_VOITURE, apportant une information sur la couleur d'une voiture, correspond à un attribut du concept C_VOITURE renseignant sur sa couleur.

Exemple 5.15 (Description du concept C_PRET)

Le concept C_PRET de notre expérimentation est défini par une règle constitutive qui possède des relations informatives avec d'autres concepts de l'ontologie. En effet le concept C_PRET est décrit par une règle sélective <C_PRET> ::= {C_PRET_AUTO | C_PRET_IMMO | C_PRET_CONSO} ;; (cf. figure 5.7) et est lié avec d'autres concepts du modèle par des relations exprimant la durée (relation R_est_durée_de avec le concept C_DUREE), le taux (relation R_est_taux_de avec le concept C_TAUX) et le montant (relation R_est_montant_de avec le concept C_SOMME) du prêt. Ces relations ne définissent pas le concept C_PRET (ni une somme, ni une durée, ni un taux ne sont un prêt) mais complètent sa description (cf. figure 5.8).

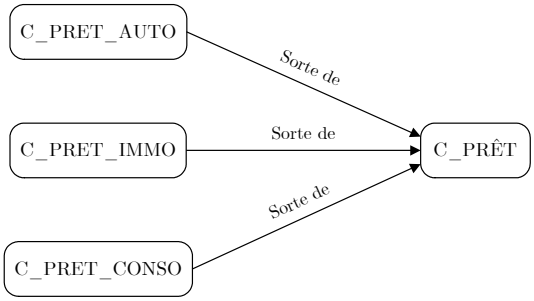

Figure 5.7 : Description du concept C_PRET

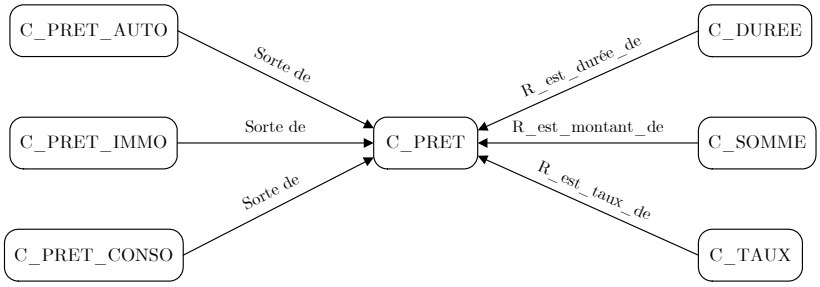

Figure 5.8 : Description du concept C_PRET et de ses relations avec d'autres concepts de l'ontologie

Les relations informatives d'un concept C_Y avec d'autres concepts du modèle sont représentées dans notre formalisme par l'ajout d'un ensemble d'*attributs* à la définition de C_Y. À l'instar des options de prédicat (cf. section 5.3.2.3), chaque attribut atY_i exprime une relation informative particulière entre C_Y et chaque élément C_A_i d'un ensemble de concepts {C_A_1, C_A_2, ..., C_A_n}. On dira que C_A_i est un type de l'attribut atY_i de C_Y. L'ensemble des types de l'attribut *at* est nommé T_atY_i (C_Y).

Un attribut atY_i est défini formellement dans une règle par un nom et la liste des éléments de T_atY_i(C_Y). L'ensemble des attributs d'un concept est exprimé par une expression entre accolades et son ajout à la définition du concept est explicité par le symbole « ++ ».

Une règle décrivant les relations informatives existant entre un concept C_Y décrit par une règle constitutive (<C_Y> ::= { β }) et d'autres concepts de l'ontologie est appelée *règle constitutive étendue* et est écrite comme suit :

<C_Y> ::= { β }
++
{
attrib_1 = <C_A1$_1$> | <C_A1$_2$> | ... | <C_A1$_{n1}$> ;
attrib_2 = <C_A2$_1$> | <C_A2$_2$> | ... | <C_A2$_{n2}$> ;
.... ;
attrib_t = <C_At$_1$> | <C_At$_2$> | ... | <C_At$_{nt}$> .
} ;;

Cette règle signifie que le concept C_Y est défini par β et qu'il existe un nombre t de relations entre C_Y et des concepts du modèle : une relation décrite par « attrib_1 » entre C_Y et chacun des concepts C_A1_1 ... C_A1_{n1}, une relation décrite par « attrib_2 » entre C_Y et chacun des concepts C_A2_1 ... C_A2_{n2}, ... et une relation décrite par « attrib_t » entre C_Y et chacun des concepts C_At_1 ... C_At_{nt} (avec t, n_i et $i \in \mathbb{N}$).

Exemple 5.16 (Règle constitutive étendue décrivant le concept C_PRET)

La règle suivante définit le concept C_PRET présenté dans l'exemple 5.15.

<C_PRET> ::= { <C_PRET_AUTO> / <C_PRET_IMMO>/
<C_PRET_CONSO> }
++
{
durée = <C_DUREE> ;
taux = <C_TAUX> ;
montant = <C_SOMME>.
}
;;

Lorsque des attributs sont définis pour un concept C_B et que C_B ∈ T_Objet(P_A), un attribut *atB* de C_B peut être considéré identique à une option *opA* de P_A (car ils expriment tous deux une relation informative entre concepts). Dans ce cas, les conditions présentées à la section 5.3.2.4 s'appliquent : même nom pour *atB* et *opA* et T_atB(C_B) ⊆ T_opA(P_A).

Considérer un attribut *atB* d'un concept C_B comme identique à une option *opA* d'un prédicat P_A permet de propager des informations d'une instance I_{C_B} de C_B à une instance I_{P_A} de P_A lorsque I_{C_B} est la valeur de l'objet de I_{P_A}.

5.4.2 Spécification des concepts génériques

Cette phase définit les concepts qui sont généralement lexicalisés dans les Notes de Communication Orale par des expressions numériques (cf. section 2.2.4.3) : le concept exprimant la notion de date ainsi que les concepts exprimant une mesure (durée, distance, poids, taux, montant, etc.).

Il s'agit de concepts communs à la plupart des modèles et des systèmes d'Extraction d'Information. Ils correspondent à une partie des entités définies lors des conférences MUC : les expressions temporelles (TIMEX) comme les dates, et les expressions numériques (NUMEX) comme les valeurs monétaires, les distances, les poids ou encore les pourcentages [Chinchor 1997]. Du fait de leur généricité, ces concepts pourront être réutilisés dans les différentes ontologies à construire en fonction des applications.

À la différence des autres concepts définis dans l'ontologie des besoins, la description de ces concepts génériques fait intervenir des informations sur leurs représentations lexicales dans les textes. En effet les productions de chacun de ces concepts (c'est-à-dire les éléments de leur extension) sont traduites dans les textes par des schémas lexicaux communs à la plupart des textes et qu'il est possible de spécifier *a priori*. Il ne s'agit pas ici d'exprimer totalement les productions lexicales de chacun des concepts génériques mais de dégager des patrons lexicaux communs à ces productions.

5.4.2.1 Concepts du type mesure

Ce type de concept regroupe l'ensemble des concepts lexicalisés dans les Notes de Communication Orale par l'association d'un nombre (entier ou réel) sous format numérique et d'une unité de mesure correspondant à la notion décrite par le concept. Il s'agit par exemple des unités "*kilomètre*", "*mètre*" ou "*année lumière*" pour le concept C_DISTANCE (exprimant la notion de distance), de "*kilogramme*" ou "*tonne*" pour le concept C_POIDS (exprimant la notion de poids), "*mois*" ou "*année*" pour le concept C_DUREE (exprimant la notion de durée), "*francs*", "*euro*" ou "*livre sterling*" pour le concept C_SOMME (exprimant la notion de somme d'argent) ou du signe pourcentage ("*%*") pour le concept C_TAUX (conceptualisant la notion de taux). Le signe "*%*" n'est pas à proprement parler une unité mais est utilisé de la même façon qu'une unité pour décrire la notion de taux dans un texte.

Pour chacun des concepts de type mesure, les différentes unités possibles sont dénombrables et facilement identifiables à l'aide de dictionnaires ou de lexiques[7]. A chaque unité correspond une méta-unité, c'est-à-dire une forme unique sous laquelle est regroupé l'ensemble des formes lexicales d'une même unité (mots, logogrammes). Cette méta-unité est en général la forme standard de l'unité (abréviation officielle du Système International d'unités). Par exemple la méta-unité définie pour l'unité exprimant le kilomètre sera "*KM*" alors que les formes lexicales de cette unité dans les textes peuvent être différentes ("*km*" mais aussi "*kilomètre*", "*kmètre*", etc.). En effet les rédacteurs ne respectent par toujours les standards d'écriture des unités (c'est particulièrement vrai dans les Notes de Communication Orale).

Nous ne cherchons pas à exprimer lors de cette étape les différentes formes lexicales correspondant à une méta-unité. Nous nous limitons ici à déterminer les formes standards de chaque unité c'est-à-dire le terme usuel pour cette unité dans la langue ou les langues du corpus ainsi que son abréviation officielle. Par exemple pour la méta-unité "*KM*" nous déterminons les entités lexicales "*kilomètre*", "*kilometer*" et "*KM*". Les éventuelles autres formes lexicales d'unités utilisées dans les textes sont déterminées lors de l'étude termino-ontologique (cf. section 6.2.3.4).

Nous définissons chaque concept du type mesure par un énoncé en langue naturelle exprimant son intension, et par une expression généralisant l'ensemble des productions de ce concept dans les textes (extension du concept). Chaque expression est formée de deux éléments : l'entité lexicale NB_NUM représentant tout nombre (entier ou réel) exprimé numériquement et la liste des différentes méta-unités correspondant au concept défini par l'expression. Par exemple le concept C_DISTANCE est défini par l'énoncé « *Intervalle mesurable qui sépare deux objets, deux points dans l'espace* » et par l'expression « NB_NUM, { AL, KM, HM, DAM, M, DM, CM, µM, NM, PM } ». Une telle expression signifie que toute production lexicale du concept C_DISTANCE est formée d'un nombre sous forme numérique suivi d'une forme lexicale d'une des méta-unités de la liste (par exemple "*1256m*", "*32 kmetre*", "*300 mètres*", etc.).

Etant donné qu'il n'est pas possible de spécifier *a priori* tous les concepts de type mesure, les concepts les plus généraux (distance, poids, taux, durée, somme) sont d'abord définis. D'autres concepts de type mesure spécifiques à un domaine ou à un corpus particulier, pourront être spécifiés lors de la réalisation d'une ontologie d'extraction élaborée pour traiter des textes de ce domaine ou de ce corpus. Ils seront décrits de la même façon que les concepts les plus généraux puis seront rajoutés à la liste des concepts génériques. Ils pourront ainsi être réutilisés pour d'autres applications du même domaine.

[7] Par exemple, les documents du Bureau International des Poids et Mesures : `http://www.bipm.fr/fr/si/` .

5.4.2.2 Concept de date

Le concept de date exprime une localisation dans le temps. Une date peut être définie par un jour, un mois et une année. Dans les textes, elle est exprimée intégralement ou partiellement sous forme de valeurs numériques, d'expression en langue naturelle ou des deux ("*3 mars 2004*"). Elle peut également être définie par une durée (la date étant alors la date courante à laquelle s'ajoute la durée, par exemple "*dans 4 ans*"), ou encore par une période ("*au 1er semestre*", "*en hiver*").

Pour prendre en compte les différentes formes de dates, le concept C_DATE est défini dans **MEGET** comme le sommet d'une hiérarchie de plusieurs sous-concepts, chacun de ces sous-concepts étant lié à C_DATE par la relation « sorte de » (relation de subsomption). Ces sous-concepts sont présentés dans la figure 5.9.

Identifiant du concept	**Type de date décrit par le concept**	**Type de date exprimé par**	**Exemples du type de date**
C_DATE_COMP	Date Complète	un jour, un mois et une anné	*04/08/2001* *5 juillet 2003*
C_DATE_SEMI_COMP	Date semi-complète	un jour et un mois	*03/05* *18 février*
C_DATE_FLOUE_1	Date floue de type 1	un mois et une année	*03/2000* *mars 1997*
C_DATE_FLOUE_2	Date floue de type 2	un mois	*novembre*
C_DATE_FLOUE_3	Date floue de type 3	une année	*en 2004*
C_DATE_FLOUE_4	Date floue de type 4	une durée fixe	*dans 4 ans*
C_DATE_FLOUE_5	Date floue de type 5	une durée floue	*dans quelques mois* *dans les mois à venir*
C_DATE_FLOUE_6	Date floue de type 6	une période fixe	*au 1er semestre* *en hiver*
C_DATE_FLOUE_IND	Date floue de type indéfini	un jour	*lundi* *le 12*

Figure 5.9 : Concepts décrivant une date

Le concept générique C_DATE est ainsi décrit par la règle sélective suivante :

<C_DATE> ::= {<C_DATE_COMP> | <C_DATE_SEMI_COMP>
| <C_DATE_FLOUE_1> | <C_DATE_FLOUE_2>
| <C_DATE_FLOUE_3> | <C_DATE_FLOUE_4>
| <C_DATE_FLOUE_5> | <C_DATE_FLOUE_6>
| <C_DATE_FLOUE_IND>
}

Nous exprimons chacun des sous-concepts de C_DATE par une expression généralisant les productions lexicales de ce concept dans les textes (cf. exemple 5.17). Cette expression est formée d'un ou de plusieurs des éléments suivants :

- Une description numérique : des nombres séparés par la barre oblique (« / »). Les nombres peuvent être exprimés directement ou par un intervalle [a,b] ;
- Un ensemble de termes choisis *a priori* (noms de jour, noms de mois, noms de saison). Ces termes peuvent être regroupés par catégorie, chaque catégorie étant nommée par un identifiant. Les identifiants de catégories pourront être utilisés dans les expressions exprimant les concepts. Par exemple les termes "*lundi*", "*Monday*", "*mardi*", "*Friday*", etc. peuvent être regroupés dans la catégorie « nom de jour de semaine » avec comme identifiant l'unité lexicale NOM_JOUR ;
- Une description mêlant des termes choisis *a priori* (les termes eux-mêmes ou des identifiants de catégorie de termes) et des valeurs numériques (nombre ou intervalle) ;
- Les opérateurs « $\wedge$ » (ET logique) et « $\vee$ » (OU logique).

Les productions lexicales des dates sont très liées à la langue dans laquelle sont rédigés les textes. En effet les termes et les représentations numériques d'une date peuvent varier d'une langue à une autre (entre le français et l'anglais par exemple). Lors de la définition des concepts génériques, il est possible de se limiter à la langue du corpus à traiter ou de tendre vers le multilinguisme en cherchant à exprimer les productions lexicales des dates dans des langues supplémentaires.

A l'instar des concepts du type mesure, les expressions des dates pourront être complétées (notamment en ajoutant de nouveaux termes exprimant des dates floues) lors de l'étude termino-ontologique (cf. section 6.2.3.4).

Exemple 5.17 (Expressions de dates)

A. Date Semi-Complète

- *Expression numérique :* « [1,31]/[1,12] », *deux nombres séparés par une barre oblique (le premier compris entre 1 et 31 et le second ente 1 et 12) ;*
- *Expression semi-numérique :* « [1,31] NOM_MOIS » *(un nombre entre 1 et 31 suivi d'un nom de mois) et* « NOM_JOUR [1,12] » *(un nom de jour de la semaine suivi d'un nombre entre 1 et 12) ;*
- *Expression alphabétique :* « NOM_JOUR NOM_MOIS » *(un nom de jour de la semaine suivi d'un nom de mois).*

La combinaison des expressions précédentes donne l'expression décrivant l'ensemble des productions lexicales du concept C_DATE_SEMI_COMP :

« [1,31]/[1,12] ∨ [1,31] NOM_MOIS ∨ NOM_JOUR [1,12] ∨ NOM_JOUR NOM_MOIS »

B. Date Floue de type 6

Les expressions

- « *en* [1900,2000] » *: le mot "en" suivi d'un nombre entre 1900 à 2000 ;*
- « *en* NOM_SAISON » *: "en" suivi d'un nom de saison :* "*automne* ", "*été* ", "*summer*", etc.),
- « *au* CARDINAL NOM_PERIODE » : *le mot "au" suivi d'un cardinal ("premier", "2ème", "second", etc.) puis d'un nom de période (" trimestre ", "semestre", etc.)*

décrivent les productions lexicales du concept C_DATE_FLOUE_6. Leur combinaison donne l'expression :

« *en* [1900,2000] ∨ *en* NOM_SAISON ∨ *au* CARDINAL NOM_PERIODE »

Chapitre 6

Étude termino-ontologique

Présentation

« *Une ontologie est une structuration hiérarchique d'un ensemble de termes du domaine* » [Swartout & al. 1997].

Les concepts décrivant les informations à rechercher et les textes sont liés dans notre approche grâce à l'unification de l'ontologie des besoins avec un modèle conceptuel fondé sur les termes du corpus et inspiré de la notion de ressource termino-ontologique [Bourigault & Aussenac-Gilles 2003]. Ce modèle, appelé ***ontologie des termes***, est constitué de concepts qui conceptualisent des termes extraits et choisis dans le corpus (*ontologie* issue *des termes*). L'élaboration de cette ontologie est réalisée à partir d'un ensemble de concepts de base (ou primitives conceptuelles [Nobécourt 2000]) en définissant récursivement de nouveaux concepts à partir de ces primitives (définition de réseaux de concepts). Une phase d'analyse terminologique (section 6.2) faisant intervenir des experts permet de déterminer les concepts de base. Lors de cette phase, la terminologie du modèle est obtenue grâce à l'extraction de termes à partir du corpus et l'ajout de termes provenant de documents externes. Le processus de construction de l'ontologie (résumé par la figure 6.1) est guidé par le but à atteindre par l'application, un but traduit par les concepts de l'ontologie des besoins.

Dans ce chapitre nous présentons les différentes étapes de la construction de l'ontologie des termes (sections 6.2 et 6.3) ainsi que les réflexions qui nous ont amené à choisir ces solutions (sections 6.1). Nous terminons ce chapitre en décrivant le processus d'unification des ontologies des besoins et de l'ontologie des termes (section 6.4).

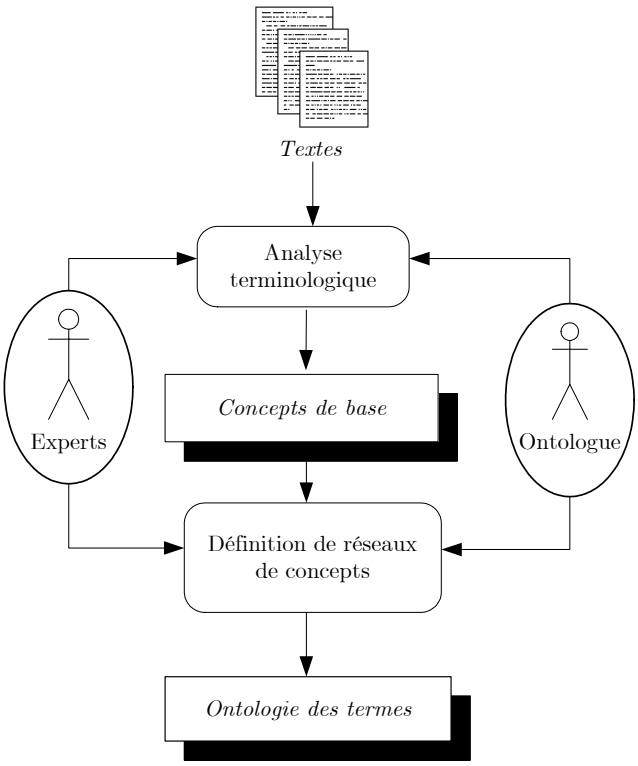

Figure 6.1 : Élaboration de l'ontologie des termes

6.1 Des concepts fondés sur des termes

6.1.1 Des termes liés à l'application

Dans notre approche, le but de l'étude terminologique du corpus est de construire une ontologie fondée sur les termes d'un corpus afin de relier les concepts d'un modèle défini en fonction d'une application spécifique (l'ontologie des besoins) avec les unités lexicales les représentant dans les textes. En conséquence la terminologie utilisée pour créer l'ontologie des termes est fortement conditionnée par les objectifs de l'application pour laquelle cette ontologie est construite, c'est-à-dire par la nature des informations recherchées et modélisées dans l'ontologie des besoins.

Cet aspect nous place dans le courant du renouvellement de la doctrine terminologique [Bourigault & Slodzian 1999] présenté dans le chapitre 4 (section 4.1) et dans la philosophie des travaux du groupe TIA[1] (Terminologie et Intelligence Artificielle). Ce point de vue nous

[1] « *Le groupe Terminologie et Intelligence Artificielle (TIA) rassemble des chercheurs en Linguistique, en Intelligence Artificielle et en Traitement Automatique des Langues. Il a été créé pour permettre une confrontation entre les cadres théoriques et méthodologiques ainsi qu'entre les pratiques développées dans chaque discipline. … cette confrontation a abouti à la mise en évidence des apports mutuels des deux disciplines (Terminologie et Intelligence Artificielle), à la suite d'une remise en cause parallèle de leurs paradigmes traditionnels. Du côté de l'Intelligence Artificielle, l'Ingénierie des Connaissances propose des concepts, outils et*

amène à considérer un terme comme une interprétation du contenu sémantique d'une unité lexicale en fonction du but de l'application. Une telle interprétation correspond à une notion (pouvant être spécifique au corpus, au domaine ou à l'application) qui est rendue explicite par un concept de base. La représentation lexicale du concept est définie par tous les termes du corpus qui expriment cette notion. Le concept peut alors être perçu comme une classe d'équivalence de termes ou plus généralement de motifs textuels. Cette définition rejoint le point de vue de Didier Bourigault et Nathalie Aussenac-Gilles pour qui un concept de base est un mode de regroupement de termes [Bourigault & Aussenac-Gilles 2003].

Nous n'avons pas ici pour volonté d'exprimer universellement tous les termes qui correspondent à chaque notion mais seulement ceux qui nous intéressent, c'est-à-dire qui peuvent être présents dans les textes et pertinents pour l'application. Cette restriction nous impose des contraintes dans le choix des termes conformes aux contraintes de pertinence décrites par Didier Bourigault et Monique Slodzian [Bourigault & Slodzian 1999] pour la construction de ressources terminologiques :

- Se focaliser selon un point de vue unique pour éliminer les contre-sens dus à des ambiguïtés sémantiques : de nombreux termes de la langue sont pluri-sémiques et décrivent des notions différentes selon le domaine dans lequel ils sont utilisés. La nécessité d'associer un concept de base unique à un terme apparaît alors contradictoire avec le pluralisme sémantique de certains termes lorsqu'ils ne sont pas pris du point de vue d'un domaine particulier ou d'une application particulière. En nous restreignant à un seul point de vue, nous limitons le nombre de signifiés de chaque terme et évacuons en conséquence une grande partie des problèmes d'ambiguïté ;
- Ne pas prendre en compte les termes inutiles à l'application : ici le but de l'ontologie est d'être une source pour un processus d'extraction, aussi seuls les termes qui ont un lien avec les connaissances recherchées nous intéressent. Il est donc inutile de retenir d'une part des termes du domaine dont il est avéré qu'ils ne seront jamais présents dans les textes car ceux-ci ne pourront pas être utiles lors de la phase d'extraction, et d'autre part des termes du corpus n'ayant aucune relation avec les éléments recherchés.

méthodes permettant d'acquérir et de modéliser des connaissances. Lorsque ces connaissances sont exprimées dans des textes, leur modélisation sollicite des méthodes et outils élaborés par la Terminologie et la Linguistique. Du côté de la Terminologie, les besoins et questionnements exprimés par l'Ingénierie des Connaissances conduisent à redéfinir le cadre théorique de la terminologie « classique » pour placer l'analyse linguistique de textes techniques au cœur de l'activité terminologique. ». Le texte précédent est tiré d'une présentation (`http://tia.loria.fr/presentation.html`) disponible en ligne sur le site internet du groupe TIA : `http://tia.loria.fr/`.

6.1.2 Méthode d'élaboration des concepts de base

Les concepts de base sont définis lors d'une phase appelée *analyse terminologique* (cf. figure 6.1). Cette analyse, décrite dans la figure 6.2, se déroule en plusieurs étapes (cf. section 6.2) :

- Une phase de prétraitements linguistiques sur le corpus visant à normaliser partiellement le corpus ;
- Une étape d'extraction de candidats-termes à partir du corpus : suite à l'étude des différentes méthodes d'extraction de terminologie (cf. chapitre 4), cette étape est réalisée avec le logiciel ANA [Enguehard & Pantéra 1995] ;
- Une étape de sélection des termes : avec l'aide des experts nous retenons comme termes les candidats-termes du corpus jugés pertinents vis-à-vis du but (et par conséquent du modèle construit en fonction de ce but). D'autres termes issus de ressources linguistiques référençant la terminologie propre au domaine du corpus (lexique, document technique, etc.) peuvent être ajoutés (cf. section 6.2.2.2) ;
- Une phase de conceptualisation des termes : nous définissons les concepts de base à partir de l'ensemble des termes en regroupant sémantiquement certains termes et en associant à chacun de ces regroupements un concept de base. Cette phase est réalisée avec l'aide d'experts et en ayant éventuellement recours à des documents externes au corpus.

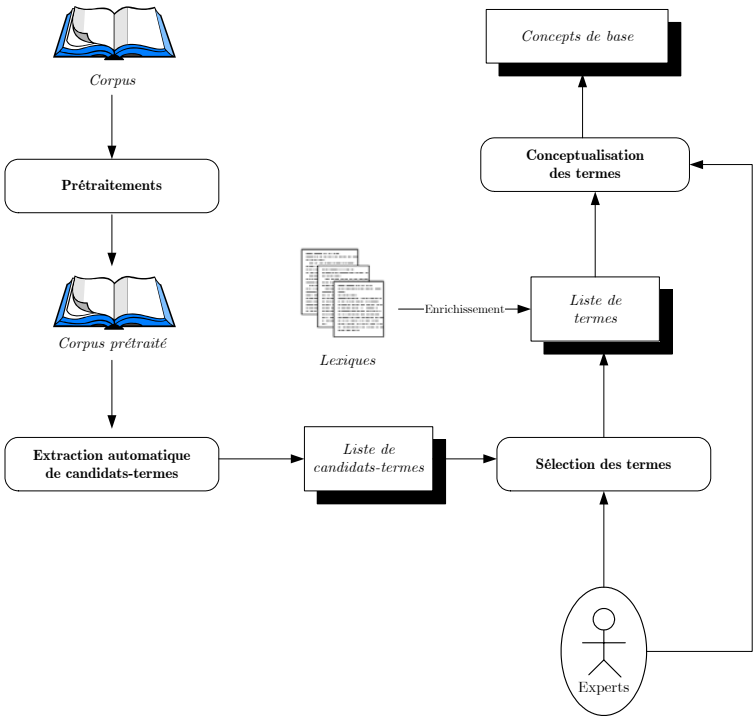

Figure 6.2 : Analyse terminologique

6.2 Analyse terminologique

Les concepts de base sont définis à partir d'un ensemble de candidats-termes obtenu par une étude terminologique des textes par le logiciel ANA et un ensemble de documents externes aux textes. Des regroupements sémantiques sont effectués sur ces termes avant de lier chacun de ces regroupements à un concept dit de base.

Avant d'exécuter le processus d'extraction de candidats-termes, nous procédons à une phase de prétraitements sur le corpus afin de corriger certains problèmes lexicaux récurrents dans les Notes de Communication Orale, des problèmes qui ne peuvent être ignorés pour s'assurer de la fiabilité du résultat de l'acquisition.

6.2.1 Prétraitements

D'après Didier Bourigault une étape de prétraitement est généralement nécessaire dans le cadre de l'extraction de terminologie car, en raison de leur variété, les corpus traités comportent fréquemment des séquences non-textuelles [Bourigault & Jacquemin 2000]. L'étude de textes possédant des particularismes linguistiques pouvant nettement nuire à la qualité des résultats renvoyés par l'analyse terminologique renforce la nécessité d'un recours préalable à un ensemble de prétraitements.

6.2.1.1 Méthodologie

Nous ne cherchons pas ici à rendre syntaxiquement ou lexicalement correctes les Notes de Communication Orale mais simplement à traiter certains problèmes récurrents dus aux particularités typographiques et morphologiques de ce type de textes (cf. section 2.2.2). Le but des ces prétraitements est d'améliorer les résultats de l'extraction de candidats-termes et, ultérieurement, ceux des phases de sélection et de conceptualisation des termes. D'une part il s'agit de corriger des problèmes inhérents au formatage du texte et aux expressions numériques (cf. section 2.2.4.3). D'autre part il s'agit de traiter le problème posé par la présence dans les textes de différentes abréviations abrégeant un même mot. Dans ce dernier cas l'objectif est d'améliorer la qualité des listes de candidats-termes extraits en normalisant les différentes représentations lexicales d'un même candidat-terme dans le corpus. En effet dans un texte chaque abréviation correspondant à un terme, une normalisation de ces abréviations permet d'éviter l'identification de plusieurs unités lexicales comme autant de candidats-termes différents alors qu'elles ne sont en réalité que des représentations lexicales d'un même candidat-terme.

La méthode de correction utilisée n'est pas spécifique à un corpus particulier. Elle est fondée sur un ensemble de règles contextuelles de réécriture écrites manuellement après l'étude d'un échantillon représentatif du corpus. Ces règles rendent compte des différentes manifestations dans les textes de trois types de problèmes évoqués précédemment (formatage, expressions numériques, abréviations). Les prétraitements sont réalisés par un module du système **SYGET** (module de prétraitements) qui applique automatiquement l'ensemble des règles contextuelles de réécriture sur les textes du corpus. Ce module est

décrit en détail dans le chapitre 7 (section 7.1). Le résultat de l'étape de prétraitements est un corpus dit *prétraité* sur lequel sera réalisée l'extraction de candidats-termes.

La réalisation de cette phase de prétraitement du corpus a d'abord été motivée par le besoin de fournir un texte partiellement « nettoyé » au processus d'acquisition de candidats-termes. Toutefois, l'ensemble de ces prétraitements nous est également apparu comme un préalable intéressant en vue du processus d'extraction à venir. En effet le travail d'extraction sur des Notes de Communication Orale partiellement normalisées produira *a priori* de meilleurs résultats que sur les textes bruts.

6.2.1.2 Limitations

Le processus de prétraitements ne cherche pas à effectuer une correction approfondie des textes mais seulement à résoudre des problèmes bien identifiés risquant de fortement dégrader les résultats de l'analyse terminologique du corpus ainsi que ceux du processus d'extraction. Cette limitation du processus s'explique par la quasi-impossibilité d'aller plus loin dans le traitement de Notes de Communication Orale. La correction syntaxique des corpus de Notes de Communication Orale est inenvisageable en raison de la multitude des problèmes de syntaxe d'un corpus à un autre et/ou d'un rédacteur à un autre.

Les fautes lexicales paraissent peu spécifiques à un corpus et leurs types assez bien définis [De Bertrand de Beuvron 1992]. Néanmoins là aussi, le traitement de ces erreurs dans les Notes de Communication Orale ne peut être envisagé. En effet la correction orthographique est un processus très lourd, quasi-inapplicable sur des Notes de Communication Orale et qui se révèle très peu généralisable à plusieurs corpus de domaines différents. La plupart des techniques mises en œuvre pour résoudre les problèmes orthographiques sont fondées sur des lexiques ou des analyseurs lexicaux, syntaxiques et grammaticaux. Leur utilisation s'avère peu efficace du fait des problèmes syntaxiques des Notes de Communication Orale. D'autres techniques fondées sur des mesures statistiques comme les calculs de similitudes et de distances entre les mots, seraient envisageables mais le choix de recourir à un extracteur terminologique comme ANA, intégrant de telles mesures, rend inutile l'utilisation de ces techniques en amont.

L'incapacité de procéder à une correction automatique avancée des Notes de Communication Orale est un argument supplémentaire pour justifier l'élaboration de techniques spécifiques non fondées sur des méthodes classique de Traitement Automatique des Langues.

6.2.2 Extraction et sélection des termes

6.2.2.1 Détermination des termes

Nous appliquons le logiciel d'extraction terminologique ANA sur le corpus prétraité ou sur un échantillon représentatif de celui-ci lorsque le corpus est de taille très importante (plusieurs dizaines de millions de mots). Dans ce dernier cas nous ne travaillons pas sur la

totalité du corpus pour améliorer les coûts des processus d'extraction et de sélection des termes.

Le résultat de l'application d'ANA est un ensemble de candidats-termes à partir desquels nous effectuons avec les experts un travail de sélection. Les candidats-termes se présentent sous la forme d'une liste de couples (*candidat-terme, fréquence dans le corpus*) triés par ordre alphabétique. Cette liste est étudiée pas à pas par l'ontologue et les experts.

Un premier filtrage est effectué en éliminant d'abord les parasites (expressions composées de caractères non-alphanumériques ne pouvant être considérées comme des termes) et les candidats-termes dont la fréquence dans le corpus est extrêmement faible. Ensuite nous procédons à un rassemblement de certains candidats-termes lorsqu'ils s'avèrent n'être que des variantes lexicographiques d'un même terme. Sont également éliminées lors de ce premier filtrage, les surcompositions (candidats-termes complexes formés à partir d'un autre candidat-terme) non pertinentes.

La sélection s'applique sur les candidats-termes issus du premier filtrage. Seuls les termes considérés comme pertinents pour l'application sont conservés. Il s'agit des candidats-termes qui, en fonction d'intuitions de la part de l'ingénieur ou des experts, paraissent liés aux informations à rechercher (informations formalisées par les concepts de l'ontologie des besoins). Concrètement un candidat-terme apparaît lié à un concept C_Y de l'ontologie des besoins dans l'un des trois cas suivant :

- Il lexicalise le concept C_Y, puisqu'il fait partie de la terminologie de C_Y. Par exemple les termes "*auto*" et "*voiture*" seront retenus car ils lexicalisent le concept C_VOITURE ;
- Il correspond à un élément de l'extension de C_Y, car il s'agit d'une production lexicale de C_Y dans le monde décrit par le corpus. Par exemple le terme "*assurance habitat personnalisée*" sera retenu comme production lexicale du concept C_ASSURANCE ;
- Il exprime une notion dont la conceptualisation peut être liée à C_Y lorsque l'ontologue possède l'intuition que le concept lexicalisé par le candidat-terme peut être lié à C_Y de manière directe ou indirecte (à travers d'autres concepts via les relations de subsomption et d'association). Par exemple le terme "*motoculteur*" lexicalise le concept C_MOTOCULTEUR qui peut être lié au concept C_VEHICULE à travers les concepts C_VEHICULE_AGRICOLE et C_DIVERS_VEHICULE en utilisant la relation de subsomption.

6.2.2.2 Enrichissement de la terminologie

Au premier ensemble de termes issus de l'extraction de candidats-termes par ANA, peuvent s'ajouter d'autres termes dits *externes* exprimant un concept donné de manière unique dans le contexte du corpus et dont nous avons une connaissance externe à une étude de surface des textes. Cette connaissance peut être issue de celle des experts du domaine mais surtout de ressources terminologiques externes référençant une terminologie spécifique

au domaine du corpus. Il peut s'agir de documents, de dictionnaires ou lexiques d'entreprise, de documents techniques ou encore d'index donnant les termes correspondant à des concepts assez généraux dans le contexte du corpus (par exemple une liste spécifique des villes de Loire-Atlantique décrivant les représentations lexicales du concept C_VILLE dans le contexte d'un corpus traitant d'opérations immobilières en Loire-Atlantique). Comme pour les candidats-termes de la section précédente, seuls sont choisis les termes externes jugés pertinents vis-à-vis de l'application par l'ontologue et les experts.

L'intégration de termes externes est essentielle car le corpus ne peut représenter tout le domaine. Afin de rendre l'application capable de traiter un corpus étendu recouvrant l'ensemble des phénomènes attendus, il faut ajouter un ensemble de termes supplémentaires. Le but est de reconnaître les concepts de base dans toutes leurs lexicalisations. Cet apport vise principalement les entités nommées mais peut recouvrir d'autres classes de termes. Par exemple dans l'expérimentation sur le corpus [CREC], les noms de tous les produits d'assurance de l'organisme bancaire (correspondant aux lexicalisations du concept C_PRODUIT_ASSURANCE) nous sont fournis dans un lexique provenant de documents internes à la banque. Il en est de même pour la liste des noms des clients de la banque.

L'ensemble de termes issus de l'analyse terminologique et de la phase d'enrichissement par des termes externes forme la *terminologie du modèle* à partir de laquelle est construite l'ontologie des termes.

6.2.3 Conceptualisation des termes

Dans cette étape nous cherchons à associer un concept de base à chacun des termes de la terminologie du modèle. Il s'agit de définir les concepts à partir desquels se construit l'ontologie des termes mais également de lier les concepts descripteurs (de prédicat ou de concept, cf. sections 5.3.2.1 et 5.4.1.1) à leurs représentations lexicales dans le corpus. Cette phase participe également à l'enrichissement des concepts génériques afin de les adapter au mieux au corpus.

6.2.3.1 Règles sélectives terminales

Une règle formelle définissant un concept par une liste des termes issus de la terminologie du modèle est une *règle sélective terminale*.

Lorsqu'un concept C_Y conceptualise l'ensemble de termes t_1, t_2, …, t_n, ce concept est décrit par la règle suivante :

<C_Y> ::= { t_1 | t_2 | t_3 | … | t_n } ;;

où t_1, t_2, …, t_n sont des termes et $n \in \mathbb{N}^+$.

6.2.3.2 Définition des concepts de base

Nous définissons les concepts de base à partir de la terminologie du modèle en regroupant sémantiquement certains termes. À chaque terme est associé un unique concept décrivant sa signification dans le contexte du domaine et de l'application, c'est-à-dire la signification qui correspond à la notion exprimée par le terme et pour laquelle il a été retenu. Cette notion correspond à l'intension du concept de base. Un concept de base conceptualise l'ensemble des termes exprimant cette notion dans le corpus. Ces termes correspondent à des productions lexicales du concept et/ou à des éléments de sa terminologie. Chaque concept de base est nommé par un identifiant (formé d'un terme préfixé par C_) exprimant son intension, préférentiellement de manière explicite.

Tous les concepts de base sont décrits par des règles sélectives terminales.

Exemple 6.1 (Règle décrivant un concept de base)

Les termes "studio", "appartement", "T1", "F4" expriment dans la notion d'appartement. Le concept C_ APPARTEMENT conceptualise cette notion et est défini par la règle :

< C_ APPARTEMENT> ::= { studio / appartement / T1 / F4 } ;;

6.2.3.3 Définition des concepts descripteurs

Lorsqu'un terme est retenu parce qu'il appartient à la terminologie d'un concept C_Y de l'ontologie des besoins défini avec un descripteur C_DC_Y (descripteur de prédicat ou de concept), le concept C_DC_Y est choisi comme concept de base de ce terme. C_DC_Y est ainsi intégré à l'ontologie des termes mais ne pourra être lié à aucun autre concept de cette ontologie.

Lorsqu'un terme est retenu parce qu'il appartient à la terminologie d'un concept C_Y décrit sans descripteur, l'ontologue peut faire le choix de définir pour C_Y un concept descripteur afin de dissocier dans l'ontologie des termes les éléments de la terminologie de C_Y de ses productions lexicales. Par exemple les termes "*region*", "*land*" sont des éléments du vocabulaire du concept C_REGION alors que les termes "*bretagne*", "*aquitaine*", "*ile de france*" en sont des productions lexicales. Si l'ontologue désire séparer les références lexicales à la notion de région et les noms de région, il définira le concept descripteur C_DC_REGION afin de conceptualiser les éléments de la terminologie de C_REGION. Les termes "*bretagne*", "*aquitaine*", etc. seront alors conceptualisés par un concept de base C_NOM_REGION.

Une règle sélective terminale décrit chaque descripteur présent dans l'ontologie des termes (cf. exemple 6.2).

Exemple 6.2 (Règle décrivant un concept descripteur)

Le concept C_DESCRIPTEUR_ACHAT (cf. section 5.3.2.1) est défini après l'étude de la terminologie du modèle par la règle sélective terminale suivante :

< C_DESCRIPTEUR_ACHAT>::= { achat / rachat / acquisition / acquerir } ;;

6.2.3.4 Enrichissement de la description des concepts génériques

Certains candidats-termes peuvent avoir été sélectionnés comme termes parce qu'ils lexicalisent une notion décrite par un concept générique, c'est-à-dire un concept du type mesure ou un des sous-concepts de C_DATE (cf. section 5.4.2).

Il s'agit de termes du corpus qui correspondent à des formes lexicales d'unités de mesure ou qui sont utilisés pour décrire un type de date (nouvelles formes de dates floues par exemple). Lorsque de tels termes sont identifiés, ils viennent enrichir les expressions décrivant les concepts génériques ("*KILOMETRE*", "*KMETRE*", "*KMS*" pour "*KM*", "*E*", "*EURO*", "*EU*" ou "*EUR*" pour "€", etc.).

6.3 Définition de réseaux de concepts

Dans cette phase, de nouveaux concepts sont définis récursivement de manière ascendante à partir des concepts de base. Ce processus vise à relier les concepts de base (et en conséquence les termes du corpus) aux concepts de l'ontologie des besoins. Ainsi les nouveaux concepts sont définis pour former des réseaux de concepts contenant au moins un concept de l'ontologie des besoins.

Ces définitions sont réalisées par l'ontologue et les experts en utilisant les règles constitutives détaillées dans le chapitre 5 (cf. section 5.4.1) mais également en étudiant le réseau sémantique fourni par ANA, l'extracteur de terminologie utilisé (cf. section 4.3.2). Les liens décrits dans ce réseau sémantique entre un terme composé et ses composés, traduisent des relations entre les concepts : des liens entre les concepts de base conceptualisant le composé et ses composants, ainsi qu'éventuellement des relations entre d'autres concepts de l'ontologie des termes liés à ces concepts de base. Il s'agit de relations qui peuvent être décrites par des règles constitutives et intégrées à l'ontologie des termes si elles sont jugées pertinentes par l'ontologue et/ou les experts.

Le résultat de cette étape est un ensemble de réseaux de concepts décrits par des règles. L'ensemble de ces réseaux forme l'ontologie des termes.

Remarque

Dans **MEGET**, nous n'avons pas recours à des thésaurus généraux comme *WordNet* [Felbaum 1998] qui décrivent des relations (hyperonymie, hyponymie, etc.) entre les termes, et *de facto* entre les concepts auxquels ils correspondent, et qui pourraient contribuer à établir des liens entre les concepts et à ajouter de nouveaux concepts à l'ontologie.

D'une part, les termes de nos corpus peuvent être absents d'un thésaurus général. Et lorsqu'il s'agit de termes courants, ils peuvent être utilisés avec un sens particulier différent de leur sens commun (par exemple dans les expérimentations sur le corpus [CREC], le terme "*orchidée*" ne désigne pas une plante mais un plan d'épargne).

D'autre part, les hiérarchies décrites dans un tel thésaurus sont trop générales en regard de l'objectif de l'ontologie. En effet l'ontologie est construite dans un but précis et est constituée de concepts et de relations en rapport avec ce but.

Le recours systématique à des thésaurus particuliers à un domaine a également été abandonné devant la difficulté de s'en procurer pour tous les domaines et/ou tous les corpus. Toutefois, il est toujours possible d'intégrer de telles ressources, quand elles existent et quand elles concernent des concepts de base, lors du processus d'enrichissement de la terminologie.

6.4 Unification des modèles

L'ontologie des termes est unifiée avec l'ontologie des besoins afin de produire l'ontologie d'extraction. La phase d'unification se déroule en deux étapes : une étape de fusion de concepts et une étape de réécriture.

En raison de la méthode de sélection des termes et de définition des hiérarchies de concepts, à chaque concept de l'ontologie des besoins correspond forcément un concept de l'ontologie des termes. Aussi l'unification est d'abord réalisée en fusionnant les concepts de l'ontologie des besoins avec les concepts de l'ontologie des termes qui leur correspondent. Lorsqu'un concept C_T de l'ontologie des termes exprime une notion conceptualisée dans l'ontologie des besoins par un concept C_B, ces deux concepts sont fusionnés. Concrètement cette fusion s'opère en remplaçant, s'ils sont différents, l'identifiant de C_T par celui de C_B.

L'étape de réécriture a simplement pour but de différencier des concepts ayant le même identifiant mais recouvrant des notions différentes : si un concept C_T' de l'ontologie des termes possède le même identifiant qu'un concept C_B' de l'ontologie des besoins mais exprime une notion différente de celle de C_B', l'identifiant de C_T' sera remplacé par un nouvel identifiant (c'est-à-dire un terme non utilisé comme identifiant d'un concept de l'une des deux ontologies).

Ces deux étapes rassemblent les règles décrivant les deux ontologies en un seul ensemble. Ce nouvel ensemble de règles décrit l'ontologie d'extraction.

La phase d'unification aboutit ainsi à la définition d'un modèle (l'ontologie d'extraction) représenté par un ensemble de règles qui formalise de manière détaillée toutes les informations à rechercher ainsi que leurs lexicalisations dans le corpus. L'ensemble de ces règles constitue la base de connaissances du système d'extraction **SYGET** présenté dans le chapitre suivant.

Chapitre 7

Le système d'extraction SyGET

Présentation

Le système **SyGET** (Système Générique d'Extraction d'informations à partir de Textes) est un système d'extraction qui extrait dans un corpus les instances des concepts décrivant les informations à rechercher. Il procède en repérant les concepts d'une ontologie écrite en utilisant la méthode et le formalisme de représentation décrits dans les deux chapitres précédents (ontologie d'extraction). L'ensemble des règles représentant l'ontologie joue le rôle de base de connaissance pour le système. Celui-ci analyse le texte en y posant des balises qui marquent les instances des concepts de l'ontologie. Le résultat est un ensemble de fichiers au format XML. Les informations sont ensuite extraites via les balises.

SyGET peut travailler sur tout type de textes et n'a pas recours à des ressources linguistiques telles que les thésaurus, les grammaires de langue ou les corpus d'apprentissage. Il se compose de quatre modules : un module de prétraitements (section 7.1), un module de génération de la base de règles (section 7.2), un module d'étiquetage (section 7.3) et un module de recueil des informations (section 7.4). L'architecture du système **SyGET** est résumée par la figure 7.1.

7.1 Module de prétraitements

Le module de prétraitements permet de résoudre certains problèmes inhérents aux Notes de Communication Orale au moyen d'un ensemble de règles contextuelles de réécriture (cf. exemple 7.1). Il s'agit de diminuer la présence de problèmes bien identifiés liés aux caractéristiques typographiques et morphologiques de ce type de texte (cf. sections 2.2.3 et 2.2.4). Ces problèmes, cités dans la section 6.2.1, concernent le formatage du texte, les expressions numériques et les abréviations.

Ces prétraitements interviennent lors de l'analyse terminologique du corpus (cf. section 6.2) dont ils améliorent la qualité, et lors de l'identification des instances des concepts de l'ontologie d'extraction (module d'étiquetage, cf. section 7.3).

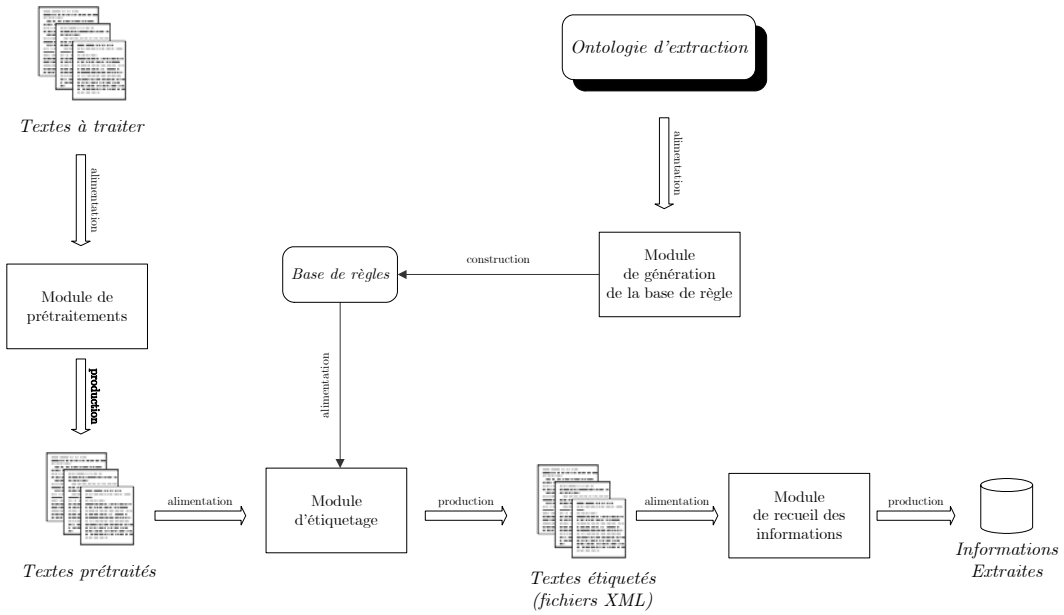

Figure 7.1 : Architecture du système **SYGET**

Le module de prétraitement utilise des règles de réécriture écrites manuellement à partir d'un échantillon représentatif du corpus à traiter. Une étude de surface de cet échantillon permet de cerner les différentes manifestations des problèmes à traiter.

Le caractère discriminant de l'étude de surface est la fréquence d'apparition d'un problème : si un problème s'avère récurrent, il apparaît nécessaire de s'en préoccuper. Cependant ce choix prend également en compte le but de l'application afin de ne pas se préoccuper de résoudre des problèmes dont la correction n'apportera aucun gain à notre application.

Une fois l'analyse terminée, ces règles sont enregistrées dans un fichier (« `prétraitements` ») que le module de prétraitements utilisera. Ces règles visent à résoudre trois types de problèmes : le formatage du texte, la normalisation des expressions numériques et la normalisation des abréviations (cf. figure 7.2).

Nous présentons d'abord dans cette section le formalisme des règles contextuelles de réécriture puis nous détaillons les différents types de règles mises en œuvre dans le module de prétraitements. Les exemples de prétraitements présentés sont ceux définis pour le corpus [CREC] écrit en français (cf. chapitre 8).

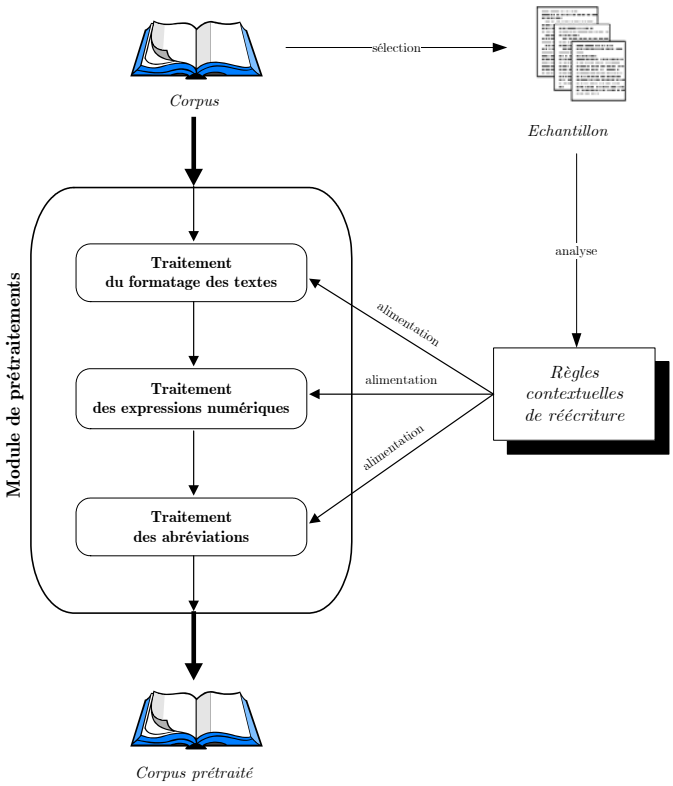

Figure 7.2 : Module de prétraitements

Exemple 7.1 (Exécution du module de prétraitements)

L'extrait de corpus :

```
PJT ACHT 04-03 BMW SERIE3 30 KEUR
```

devient, après l'exécution du module de prétraitements, le texte suivant :

```
projet achat 04/2003 bmw serie 3 30ke
```

en utilisant les règles

```
' ([0..9])([a..z]) ' -> ' $1 $2 '
' PJT ' -> ' projet '
' ([01..31])-([01..09]) ' -> ' $1/20$2 '
' ACHT|ACH ' -> ' achat '
' ([0..9]+)( )*(KEUR|KEU|KILOEURO|KEURO) ' -> ' $1ke '
```

7.1.1 Formalisme de réécriture

Les règles contextuelles de réécriture (cf. exemple 7.2) sont composées d'un membre gauche (MG) et d'un membre droit (MD) décrits chacun entre « ' » et séparés par le symbole « -> ». Elles sont ainsi de la forme `'MG' -> 'MD'.` Le membre gauche met en jeu des unités lexicales (mots, caractères) ainsi que des expressions mêlant des unités lexicales et des éléments dont la syntaxe s'inspire de celle des expressions régulières du langage Perl [Wall & al. 2000] (syntaxe connue et simple à utiliser). Les différents éléments du membre gauche d'une règle peuvent être placés entre parenthèses. Le membre droit est composé de termes et/ou de variables nommées `$i` avec i $\in \mathbb{N}*$.

Dans une expression, les éléments suivants ont une signification particulière :

- Le symbole « | » correspond au OU exclusif ;
- `[a..b]` décrit l'intervalle de `a` à `b` ;
- Les symboles « `+` », « `*` » et « `?` » derrière une expression ou une unité lexicale possèdent les mêmes significations que dans les expressions régulières (respectivement au moins une fois, 0 à n fois avec $n \in \mathbb{N}$, 0 ou 1 fois) ;
- Le caractère espace représente un ou plusieurs espaces ;
- Une variable `$i` dans le membre droit d'une règle correspond à la valeur de la $i^{ème}$ paire de parenthèses de son membre gauche. La valeur d'une paire de parenthèses correspond aux éléments lexicaux contenus dans l'expression entre parenthèses et repérées dans le texte.

Les symboles ayant une signification particulière sont appelés méta-symboles. L'ajout du symbole « \ » devant un méta-symbole dans une règle signifie qu'il y est utilisé comme un caractère textuel et non comme un méta-symbole.

Dans **SYGET**, une règle contextuelle de réécriture est appliquée en remplaçant chaque occurrence de la séquence lexicale décrite par son membre gauche (morceau de texte trouvé dans le corpus) par son membre droit (morceau de texte corrigé).

Exemple 7.2 (Règles contextuelles de réécriture)

```
1.   ' JEU ([01..31]/[01..31])' -> ' jeudi $1'
2.   ' (CTE|CPTE|CPT) ' -> ' compte '
```

L'application de la règle 1 remplace les occurrences du mot "JEU" par "jeudi" lorsqu'elles sont suivies d'une expression composée de deux nombres compris entre 1 et 31 et séparés par la barre oblique ("/").

L'application de la règle 2 remplace chacune des occurrences des termes "CTE", "CPTE" et "CPT" par le mot "compte".

7.1.2 Formatage du texte

Ces prétraitements visent à supprimer les caractères de formatage présents dans les textes ainsi qu'à traiter des caractères particuliers non-alphanumériques comme les signes de ponctuation. Les caractères de formatage sont éliminés des textes alors que les caractères particuliers sont traités au cas par cas.

Le traitement des caractères particuliers consiste d'une part à isoler des mots, la ponctuation et certains caractères spécifiques comme les signes de ponctuation, les apostrophes ou les guillemets, les opérateurs (“+”, “-”, ...) ou d'autres signes mathématiques (“<”, “=”, ...) en ajoutant un espace lorsque c'est nécessaire.

Dans cette étape, nous séparons également les nombres des mots en insérant un espace entre un chiffre et une lettre (par exemple l'unité lexicale “*serie3*” est transformée en “*serie 3*”, cf. exemple 7.1).

7.1.3 Expressions numériques

Les expressions numériques regroupent les expressions arithmétiques (nombres, opérations), les expressions formées d'une valeur numérique et d'une unité (expression exprimant une mesure) et les dates. En dehors des expressions arithmétiques, les expressions numériques correspondent aux productions lexicales des concepts génériques présentés dans le chapitre 5 (section 5.4.2) : les concepts du type mesure et le concept de date.

7.1.3.1 Expressions arithmétiques

Les expressions arithmétiques comprennent les nombres entiers ou décimaux, mais également les opérations entre nombres :

- Tous les nombres décimaux sont normalisés sous la forme de deux suites de chiffres séparés par une virgule grâce à la règle de réécriture :

```
' ([0-9]+) ([.,]) ([0-9]+) ' -> '$1,$3'
```

- Chaque opération est transformée pour n'apparaître que sous un format unique. Ici le terme *opération* englobe les quatre opérations mathématiques classiques (addition soustraction, multiplication et division) ainsi que les opérations de comparaison (inférieur, supérieur, égale, etc.).
  Pour chaque caractère représentant un opérateur placé entre deux suites de chiffres s'applique une règle de réécriture renvoyant comme résultat une unité lexicale composée de la première suite de chiffres suivie directement de l'opérateur puis de la deuxième suite de chiffre. Les caractères correspondant aux opérateurs sont des caractères non alphanumériques (“ * ”, “ - ”, “ + ”, “ < ”, “ > ”, etc.) ainsi que certaines lettres qui placées entre deux chiffres traduisent un opérateur (c'est le cas du caractère *X* par exemple).

Exemple 7.3 (Traitement des expressions arithmétiques)

A. Exemple de règles

```
' ([0-9]+) \* ([0-9]+) ' -> ' $1*$2 '
' ([0-9]+) x ([0-9]+) ' -> ' $1*$2 '
' ([0-9]+) < ([0-9]+) ' -> ' $1<$2 '
```

B. Exemple d'application

L'extrait de texte : « REMB DE 300 + 200 KF SUITE EMPRUNT » est transformé en « REMB DE 300+200 KF SUITE EMPRUNT » par la règle :

```
' ([0-9]+) \+ ([0-9]+) ' -> ' $1+$2 '
```

7.1.3.2 Mesures

Les expressions formées d'une valeur numérique et d'une unité (poids, taille, somme d'argent, durée, etc.) sont les productions lexicales des concepts génériques de type mesure (cf. section 5.4.2.1). Afin de rendre plus efficace l'identification des instances de ces concepts dans les textes, nous procédons à une formalisation de ces productions lexicales. Le format retenu correspond à celui choisi pour décrire les concepts génériques de type mesure, c'est-à-dire une suite de chiffres suivie par une forme unique pour chaque unité (méta-unité).

Pour chaque méta-unité *MU* retenue lors de la phase de spécification des concepts de type mesure, nous écrivons une règle contextuelle de réécriture dont l'application recherchera chaque séquence textuelle composée d'une forme lexicale de *MU* (formes déterminées lors de l'analyse terminologique du corpus) précédée d'un nombre *N* entier ou décimal, puis remplacera cette séquence par le nombre *N* suivi directement de la méta-unité *MU*.

Le symbole pourcentage est traité de la même façon car nous considérons le pourcentage comme la méta-unité correspondant à un taux.

Exemple 7.4 (Traitement des mesures)

Les deux extraits de texte :

1. ACHAT TWINGO 32000 KMS FINCT SUR 60 MOIS DEMANDE TX 6,6% .
2. MISE EN PLACE PRET PERSONNALISE DE 40 KEUR SUR 37 MOIS AVEC 600 EUR MENS ET 11 MOIS AVEC 2600EU.

sont respectivement transformés en :

1. ACHAT TWINGO 32000KM FINCT SUR 60MOIS DEMANDE TX 6,6% .
2. MISE EN PLACE PRET PERSONNALISE DE 40KE SUR 37MOIS AVEC 600EUROS MENS ET 11MOIS AVEC 2600EUROS.

par l'application des règles suivantes

```
' ([0-9]+) (EURO|EUROS|EUR|EU) ' -> ' $1EURO '
' ([0-9]+(,[0-9]+)?) (KILOMETRE|KILOMETRES|KMS|KMETRE|KMETRES) ' -> ' $1KM '
```

7.1.3.3 Dates

La notion de date peut être exprimée de différentes manières dans des textes : numériquement, par des mots ou de façon mixte mêlant nombres et mots (cf. section 5.4.2.2). Nous nous intéressons ici uniquement aux représentations numériques du concept C_DATE dans les textes.

Il existe de nombreuses façons de représenter numériquement des dates. Le traitement des dates consiste à formaliser ces représentations afin d'homogénéiser les informations contenues dans les textes (dans le but de simplifier la recherche ultérieure des instances du concept C_DATE). Le format retenu est un ensemble de nombres séparés par la barre oblique. Grâce à un ensemble de règles de réécriture, les suites de chiffres et de symboles particuliers (point, slash, espace, etc.) identifiées comme des représentations numériques de date sont transformées dans le format retenu (c'est-à-dire en JJ/MM/AAAA ou MM/AAAA ou encore JJ/MM selon les règles utilisées).

Les dates exprimées numériquement et de manière complète ainsi que certaines dates dont la représentation numérique est incomplète (dates semi-complètes et dates floues de type 1) sont réécrites. Il s'agit des productions lexicales des concepts C_DATE_COMP, C_DATE_SEMICOMP et C_DATE_FLOUE_1.

L'exemple 7.5 présente quelques-unes des productions de dates trouvées dans le corpus [CREC] et le résultat de leur traitement par le module de prétraitements.

Exemple 7.5 (Traitement de représentations numériques de dates)

Les extraits de texte suivants :

1. `MISE EN PLACE D'UN VIT AUTO DE 3500 A PARTIR DU 12.06.1998 (date complète)`
2. `LES ASSEDIC SERONT VERSEES VERS LE 26081999 3300 FRANCS. (date complète)`
3. `SIMUL MRH POUR 5 ETUDIANTS DANS LE MEME LOGEMENT T5 LE 0608.` (date semi-complète)

sont respectivement transformés en :

1. `MISE EN PLACE D'UN VIT AUTO DE 3500 A PARTIR DU 12/06/1998`
2. `LES ASSEDIC SERONT VERSEES VERS LE 26/08/1999 3300 FRANCS.`
3. `SIMUL MRH POUR 5 ETUDIANTS DANS LE MEME LOGEMENT T5 LE 06/08.`

par l'application des règles suivantes

```
' ([01..31]+)\.([01..12])\.([1900..2099]) ' -> ' $1/$2/$3 '  (cas 1)
' ([01..31]+)([01..12])([1900..2099]) ' -> ' $1/$2/$3 '  (cas 2)
' (le|au|du) ([01..12])([01..31]) ' -> ' $1 $2/$3 '  (cas 3)
```

7.1.4 Abréviations

Les corpus de Notes de Communication Orale contiennent de nombreuses abréviations. Ces abréviations peuvent être laissées telles quelles dans le corpus puis traitées comme n'importe quel autre mot ; mais il est possible de les normaliser en les remplaçant dans les textes par les termes qu'elles abrègent afin d'augmenter la lisibilité et l'uniformisation du corpus et améliorer ainsi les résultats des processus d'extraction de termes puis de reconnaissance des instances de concept.

Pour réaliser correctement cet objectif, il est nécessaire de résoudre les problèmes d'ambiguïté sémantique introduits par certaines abréviations dans les cas où une même abréviation ne désigne pas un terme de manière unique dans le corpus. Par exemple l'abréviation "*DEC*" peut signifier le mois de "*décembre*" ou le terme "*décroissant*" ou encore "*découvert*". Ces problèmes d'ambiguïté sont assimilables à des problèmes d'homonymie et doivent être traités lors du processus de normalisation pour éviter des erreurs importantes dues à une normalisation erronée, c'est-à-dire le remplacement d'une abréviation par un mauvais terme : erreurs d'identification de candidats-termes et changement du sens de phrases du texte. Le traitement des ambiguïtés permet également d'éviter de traiter comme des abréviations, des mots du texte lexicalement équivalents à des abréviations comme par exemple "*VERS*" qui peut être interprété comme l'abréviation du terme "*versement*" mais qui est aussi présent dans les textes comme l'adverbe de direction "*vers*".

Les cas d'ambiguïté sont résolus par une étude des différents *contextes lexicaux* de l'abréviation. Un contexte lexical correspond aux expressions lexicales (suites d'unités lexicales) situées à droite et à gauche (on parlera respectivement de contexte droit et de contexte gauche) de l'abréviation.

Les abréviations utilisées par les auteurs de Notes de Communication Orale sont variées et relativement peu ou pas normalisées (abréviations particulières à chaque rédacteur, différentes des standards habituellement utilisés). Néanmoins, nous observons qu'il n'existe qu'un nombre restreint d'abréviations apparaissant de manière récurrente dans les textes et abrégeant des mots ou termes exprimant des concepts importants d'un point de vue de l'application. Une étude de surface d'un échantillon du corpus se focalisant sur la fréquence des abréviations permet d'établir une liste de ces abréviations.

La liste des abréviations récurrentes indique, pour chaque abréviation, le terme qu'elle abrège (ce terme est identifié en étudiant la signification de l'abréviation dans le texte) et, en cas d'ambiguïté, ses contextes.

Cette liste d'abréviations permet d'établir un ensemble de règles de réécriture dont l'application sur le corpus remplace chaque abréviation du texte par le terme qu'elle abrège. Dans les cas où plusieurs termes sont abrégés par une même abréviation, leurs contextes lexicaux sont analysés afin d'établir des règles de désambiguïsation locale. Le processus appliquant les règles assure qu'une fois qu'une abréviation est remplacée par un terme, ce terme ne peut plus être interprété comme une abréviation par le processus (même si ce terme est lexicalement équivalent à une abréviation de la liste).

7.1.4.1 Règles simples

Dans les cas où il n'existe pas d'ambiguïté les règles se résument à la forme suivante (règles simples) :

```
' ABREVIATION 1 ' -> ' terme_abrégé '
```

ou

```
' (ABREVIATION 1|... |ABREVIATION n) ' -> ' terme_abrégé '   avec n > 1
```

Une règle est appliquée par le système en remplaçant dans le texte toutes les occurrences de « `ABREVIATION x` » par « `terme abrégé` ». Ci dessous sont présentés quelques exemples de règles simples.

Exemple 7.6 (Règles simples de traitement des abréviations)

```
' AOU ' -> ' AOUT '
' (SEP|SEPT) ' -> ' SEPTEMBRE '
' (CHG|CHGT|CHANGT) ' -> ' changement '
' (APP|APT|APPT|APPRT|APPART) ' -> ' appartement '
' EP ' -> ' epargne '
' (C / C|C / CH|CCH|CCHQ|C / CHQ|CCHQUE|CC) ' -> ' compte cheque '
' (CH|CHQ|CHQUE|CHEQ) ' -> ' cheque '
' (VERST|VERS|VERT) ' -> ' versement '
```

7.1.4.2 Règles de désambiguïsation locale

Pour traiter les cas où une abréviation est ambiguë, nous utilisons des règles dites *de désambiguïsation locale*. Celles-ci permettent de remplacer une abréviation ambiguë par le terme qu'elle abrège dans la portion de texte où elle est utilisée, c'est-à-dire le terme dont la signification correspond à celle choisi par le rédacteur pour cette abréviation.

Pour une abréviation, les règles de désambiguïsation locale sont écrites en étudiant ses différents contextes lexicaux dans les textes. À partir de ces contextes lexicaux, nous extrayons du corpus un ensemble de *contextes locaux* pour chaque abréviation ambiguë de notre liste. Un contexte local est formé d'unités lexicales situées à gauche et/ou à droite d'une abréviation dans le texte qui permettent à un lecteur de comprendre sans ambiguïté la signification de l'abréviation (et par conséquent de trouver le terme qu'elle abrège). La taille d'un contexte local n'est pas fixe, elle peut être plus ou moins réduite en fonction du nombre d'unités lexicales nécessaires à la compréhension de la signification de l'abréviation.

Le membre gauche d'une règle de désambiguïsation locale est formé de l'abréviation traitée par la règle et d'une suite d'unités lexicales à sa droite et/ou à sa gauche décrivant

un contexte local de l'abréviation. Le terme abrégé correspondant au contexte local de la partie gauche, se trouve dans le membre droit de la règle. Le membre droit peut également contenir d'autres termes ainsi qu'une ou plusieurs variables (cf. exemple 7.7).

Afin de ne pas multiplier le nombre de règles, une même règle de désambiguïsation locale peut traiter plusieurs abréviations abrégeant le même terme. Dans ce cas, la règle n'est pas centrée sur une seule abréviation mais sur une liste d'abréviations, le terme abrégé restant unique.

Exemple 7.7 (Règles de désambiguïsation locale)

- `' livret (B|BL) ' -> ' livret bleu '`
  "B" ou "BL" précédé de "livret" signifie "bleu". Chaque occurrence de "livret B" ou de "livret BL" sera remplacée par le terme livret "bleu".
- `' ([1..31]) DEC ' -> ' $1 decembre '`
  L'abréviation "DEC " signifie "decembre" lorsqu'elle est précédée d'un nombre entre 1 et 31. Lorsque ce cas se produit dans le texte, "DEC " sera remplacé par le terme "decembre", le nombre restant inchangé.
- `' (EN|DEPUIS|FIN|MI) DEC ' -> ' $1 decembre '`
  "DEC " signifie "décembre" lorsqu'elle est précédée de "EN " ou de "DEPUIS " ou de "FIN" ou encore de "MI"
- `' VIT (a|chez|en|depuis|desormais|actuellement) ' -> ' vit $1 '`
  "VIT " suivi d'un des mots de la liste entre parenthèses signifie "vit" (conjugaison du verbe "vivre").
- `' VIT sur ' -> ' virement sur '`
  "VIT " suivi de "sur" signifie "virement"
- `' r (secondaire|principale|locative) ' -> ' residence $1 '`
  "R" suivi de "secondaire" ou de "principale" ou de "locative" signifie "residence".

Dans certains cas le contexte local d'une abréviation est lui-même composé d'abréviations, l'application des règles contextuelles correspondantes permet alors au processus d'effectuer en une passe le remplacement de l'abréviation ambiguë ainsi que celles appartenant à son contexte local. Ci-dessous se trouvent des exemples de telles règles :

```
' (R|residence) (PRINC|PRINCIP|PPAL) ' -> ' residence principale '
' VERS (supplementaire|SUP) ' -> ' versement supplementaire '
```

Lorsqu'une abréviation ambiguë abrège n termes, il existe soit des règles de désambiguïsation locale traitant les n termes qu'elle abrège, soit des règles traitant *n-1* termes. Dans ce deuxième cas, le terme non traité est considérée comme le « *terme par défaut* » de l'abréviation. Le terme par défaut correspond au terme abrégé par l'abréviation lorsqu'elle n'apparaît dans aucun contexte local.

Exemple 7.8 (Terme par défaut d'une abréviation)

```
' versement SUP ' -> ' versement supplementaire '
' virement SUP ' -> ' virement supplementaire '
' SUP ' -> ' superieur '
```

Dans cet exemple "superieur" est le terme par défaut de "SUP".

Une abréviation n'a pas de terme par défaut quand les contextes locaux déterminés pour l'abréviation sont traités par des règles de désambiguïsation, ou quand la ou les unités lexicales ne correspondent finalement pas à une abréviation (par exemple l'unité lexicale "*VERS*" qui peut être l'abréviation du mot "*versement*" mais également la préposition "*vers*").

7.1.4.3 Application des règles

Afin de traiter correctement les abréviations, le module de prétraitements intègre des contraintes sur l'ordre d'application des règles et l'apparition de nouveaux contextes locaux dans les textes en réordonnant les règles et en en réécrivant certaines.

7.4.3.1.1 Ordre des règles

En raison de la présence éventuelle d'un terme par défaut pour certaines abréviations ambiguës, les règles simples doivent être appliquées après les règles de désambiguïsation locale.

Pour respecter cet ordre d'application, le processus sépare automatiquement les règles du fichier `« prétraitements »` en deux ensembles distincts : les règles de désambiguïsation locale et les règles simples. Il commence par appliquer les règles de désambiguïsation locale, puis, quand plus aucune d'entre elles n'est applicable (c'est-à-dire lorsque plus aucune règle n'engendre de modification du texte), les règles simples sont appliquées à leur tour.

7.4.3.1.2 Gestion de nouveaux contextes locaux

La transformation des textes par l'application des règles simples peut entraîner l'apparition de nouveaux contextes locaux pour certaines abréviations (par exemple avec l'application d'une règle transformant une abréviation ABV en un terme T, les contextes locaux contenant ABV disparaissent du texte pour être remplacés par de nouveaux contextes locaux contenant le terme T).

L'apparition de nouveaux contextes locaux par les règles simples est gérée en amont de l'exécution des règles sur le texte. Lors de la création de la base de règles, le processus réécrit automatiquement les règles de désambiguïsation locale de la manière suivante : lorsqu'un terme présent en partie gauche d'une règle de désambiguïsation locale *Rg* correspond à un terme en partie droite d'une règle simple, alors les abréviations en partie gauche de cette règle simple sont intégrées à la partie gauche de *Rg*. Par exemple le processus intègre la partie gauche de la règle simple `' (VERST|VERS|VERT) ' -> ' versement '` à la règle de désambiguïsation locale `' versement SUP ' -> ' versement supplémentaire '` et produit ainsi la

règle `'(VERST|VERS|VERT|versement) SUP ' -> ' versement supplémentaire '` qui sera placée dans la base de règles et appliquée sur le corpus.

L'automatisation de la gestion de l'apparition de nouveaux contextes locaux permet de ne pas chercher à prendre en compte ces cas souvent complexes lors de l'élaboration manuelle des règles. Elle permet alors à la phase de création des règles et aux règles elles-mêmes, de conserver un caractère simple et efficace.

7.2 Module de génération de la base de règles

Le module de génération de la base de règles transforme de manière automatique l'ensemble de règles formelles représentant l'ontologie d'extraction en une base de règles intelligibles par le module d'étiquetage. Ce module prend en entrée un fichier contenant toutes les règles formelles et produit automatiquement un ensemble de fichiers directement utilisables par le module d'étiquetage (un exemple de fichiers créés par ce module est présenté en annexe C).

Le module traite automatiquement chaque règle formelle en fonction de son type et génère quatre fichiers qui forment la base de règles du système (cf. figure 7.3) : « `règles terminales` » correspondant aux règles sélectives terminales (cf. section 6.2.3), « `règles constitutives concepts` » correspondant aux règles sélectives, disjonctives et conjonctives (cf. section 5.4.1), « `règles informatives` » correspondant aux règles constitutives étendues (cf. section 5.4.1.4) et « `règles prédicatives` » correspondant aux règles prédicatives (cf. section 5.3.3). La base de règles n'intègre pas les règles définissant les concepts génériques.

Le module permet également de paramétrer certains types de règles de la base afin d'améliorer leur adéquation avec les corpus à traiter.

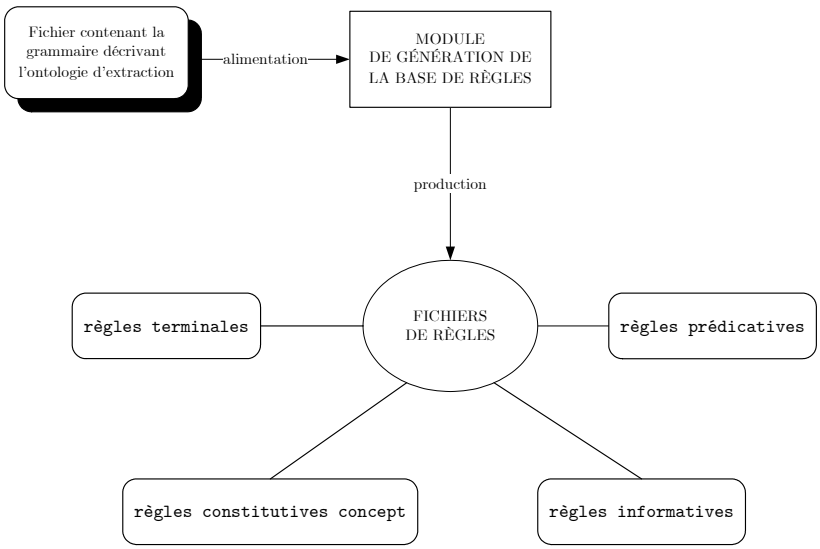

Figure 7.3 : Module de génération de la base de règles

7.2.1 Règles sélectives terminales

Chaque règle sélective terminale est transformée de manière automatique en une expression respectant le formalisme de réécriture décrit dans la section 7.1.1.

Une règle : <C_Y> ::= { t_1 | t_2 | t_3 | ... | t_n } ;; devient

' (t_1 | t_2 | t_3 | ... | t_n) ' -> ' <C_Y>$1</C_Y> '

Une telle expression signifie que la présence d'un terme t_i dans le texte détermine une instance du concept C_Y.

Exemple 7.9 (Transformation d'une règle sélective terminale)

La règle

<C_DC_PROPOSITION> ::= {proposer / propose / proposition / offre / demande} ;;

est transformée en l'expression suivante :

```
' (proposer | propose | proposition | offre | demande) ' -> '
<C_DC_PROPOSITION>$1</C_DC_PROPOSITION> '
```

L'ensemble des règles de réécriture créées à partir des règles sélectives terminales sont placées dans un fichier unique « règles terminales ».

7.2.2 Règles constitutives sur les concepts

7.2.2.1 Réécriture des règles

Les règles constitutives sont réécrites dans un formalisme propre à SYGET. Les règles transformées sont enregistrées dans un fichier « règles constitutives concept ». Chacune d'entre elles est précédée dans le fichier par un identifiant unique formé d'une catégorie et d'un numéro : la catégorie marque le type de la règle (RS pour règle sélective, RD pour règle disjonctive ou RC pour règle conjonctive) et le numéro permet d'identifier précisément une règle parmi toutes celles de sa catégorie.

Pour plus de lisibilité, les règles sont classées dans le fichier <règles constitutives concept> par type puis par numéro avec une seule règle par ligne : d'abord les règles RS (RS1, RS2, ..., RSn), puis les règles RD (RD1, RD2, ..., RDn) et enfin les règles RC (RC1, RC2, ..., RCn). On dénotera par « règle Rα » une règle de type Rα.

7.2.2.1.1 Règles sélectives

Une règle sélective <C_Y> ::= { <C_A_1> | < C_A_2> | ... | < C_A_n> };; est transformée en l'expression `C_A1 | C_A2 | ... | C_An -> C_Y` . Une telle règle est notée dans le fichier par un identifiant RSx (avec x le numéro de la règle).

Une telle expression signifie que la présence d'une instance d'un concept C_A_i dans le texte détermine une instance de C_Y.

Exemple 7.10 (Transformation d'une règle sélective)

La règle

<C_PROD_BANCAIRE> ::= {<C_ASSURANCE> / <EPARGNE>
/ <C_SERVICE_BANCAIRE>
/ <C_DIV_PROD_BANCAIRE> } ;;

est transformée en l'expression suivante :

```
C_ASSURANCE | C_EPARGNE | C_SERVICE_BANCAIRE | C_DIV_PROD_BANCAIRE
-> C_PROD_BANCAIRE
```

7.2.2.1.2 Règles conjonctives

Une règle conjonctive <C_Y> ::= { < C_A_1> + < C_A_2> + ... + < C_A_n> } ;; est transformée en l'expression `C_A1 + C_A2 +...+ C_An -> C_Y` . Une telle règle est notée dans le fichier par un identifiant RCx.

Une telle expression signifie que la présence dans le texte d'une instance de C_A_1 suivie d'une instance de C_A_2 ... suivi d'une instance de C_A_n défini une instance de C_Y. Les instances de concepts doivent apparaître dans l'ordre exprimé dans la règle (les éléments pouvant séparer ou non les instances de C_A_i dans le texte sont fixés par un paramètre dit de séparation, cf. section 7.2.2.3).

Exemple 7.11 (Transformation d'une règle conjonctive)

La règle

<C_PRET_IMMO>= {<C_DC_PRET> + <C_IMMOBILIER>} ;;

est transformée en l'expression suivante :

```
C_DC_PRET + C_IMMOBILIER -> C_PRET_IMMO                (règle RC)
```

7.2.2.1.3 Règles disjonctives

Une règle disjonctive <C_Y> ::= { < C_A_1> ∨ < C_A_2> ∨ ... ∨ < C_A_n> } ;; est transformée en deux expressions :

```
C_A1 |+ C_A2 |+ ... |+ C_An -> C_Y
```

et

```
C_A1 | C_A2 |...| C_An -> C_Y
```

La première expression utilise l'opérateur « |+ » et signifie qu'une instance de C_Y est définie par la présence d'au moins deux instances de concepts C_A_i. L'ordre d'apparition de ces instances de C_A_i dans le texte n'a ici pas d'importance (soit I_1 et I_2 respectivement des instances de C_A_1 et C_A_2, les séquences « I_1 I_2 » et « I_2 I_1 » dans le texte déterminent chacune une instance de C_Y). Cette expression est notée dans le fichier par l'identifiant RDx.

La deuxième est une règle de type RS exprimant que n'importe quelle instance de C_A_i dans le texte définit une instance de C_Y.

Exemple 7.12 (Transformation d'une règle disjonctive)

La règle

<C_AUTO>::= {<C_TYPE_AUTO>∨<C_MARQUE_AUTO>∨<C_MODELE_AUTO>} ;;

est transformée en :

```
C_TYPE_AUTO |+ C_MARQUE_AUTO |+ C_MODELE_AUTO -> C_AUTO        (règle RD)
```

et

```
C_TYPE_AUTO | C_MARQUE_AUTO | C_MODELE_AUTO -> C_AUTO          (règle RS)
```

7.2.2.1.4 Règles constitutives étendues

Les règles constitutives étendues (cf. section 5.4.1.4) décrivent un concept par une règle constitutive et expriment un ensemble de relations informatives pour ce concept.

Une règle constitutive étendue est de la forme (avec t, n_i et i ∈ ℕ) :

<C_Y> ::= { β }
++
{
attribut_1 = <C_A1$_1$> | <C_A1$_2$> | ... | <C_A 1$_{n1}$> ;
attribut_2 = <C_A2$_1$> | <C_A2$_2$> | ... | <C_A2$_{n2}$> ;
....
attribut_t = <C_At$_1$> | <C_At$_2$> | ... | <C_At$_{np}$> .
} ;;

Le module de génération de la base de règles traite une telle règle en deux étapes. La première étape traite la partie constitutive de la règle (<C_Y> ::= { β }) et la deuxième l'ensemble d'attributs :

- Une règle <C_DF_Y> ::= { β } ;; est créée à partir de la partie constitutive <C_Y> ::= { β }. Le concept C_DF_Y est un concept lié uniquement à C_Y et généré par le système. Il ne s'agit pas d'un concept présent dans l'ontologie d'extraction. Cette nouvelle règle est transformée en fonction de sa nature en suivant les principes évoqués dans les sections précédentes. Ainsi le module générera la règle

RSx β' -> C_DF_Y, s'il s'agit d'une règle sélective, la règle RCx β' -> C_DF_Y s'il s'agit d'une règle conjonctive, ou les règles RDx β' -> C_DF_Y et RSz β'' -> C_DF_Y s'il s'agit d'une règle disjonctive. Le résultat de cette transformation est placé dans le fichier « règles constitutives concept » ;

- À partir de l'ensemble d'attributs, le module génère l'expression ci-dessous (règle informative du système) et la place dans le fichier « règles informatives ».

```
C_Y  :  C_DF_Y
        C_Y_attribut_1 : C_A1_1 | C_A1_2 | ... | C_A1_n1
        C_Y_attribut_2 : C_A2_1 | C_A2_2 | ... | C_A2_n1
        ...
        C_Y_attribut_t : C_At_1 | C_At_2 | ... | C_At_n1
```

Le concept C_DF_Y est l'équivalent du descripteur dans une règle prédicative.

Dans les cas où C_Y est décrit par une règle constitutive étendue dans laquelle un (ou plusieurs) attribut est défini par :

attribut_q == <C_Aq$_1$> | <C_Aq$_2$> | ... | <C_Aq$_{nq}$>

cette information se traduira dans le fichier de règles par

C_Y_attribut_q :: C_Aq$_1$ | C_Aq$_2$ | ... | C_Aq$_{nq}$

Exemple 7.13 (Transformation d'une règle constitutive étendue)

La règle

```
<C_PRET> ::=   { <C_PRET_AUTO> / <C_PRET_IMMO>/
                 <C_PRET_CONSO> }
           ++
           {
                 durée      =   <C_DUREE> ;
                 taux       =   <C_TAUX> ;
                 montant    =   <C_SOMME>.
           } ;;
```

est transformée en :

```
C_PRET_AUTO | C_PRET_IMMO | C_PRET_CONSO -> C_DF_PRET        (règle RS)
```

et

```
C_PRET :    C_DF_PRET
            C_PRET_durée   : C_DUREE
            C_PRET_taux    : C_TAUX
            C_PRET_montant : C_SOMME
```

7.2.2.2 Résolution des conflits entre règles

Avant de pouvoir appliquer les règles du fichier « règles constitutives concept », il convient de vérifier s'il existe des conflits entre certaines règles et, dans ce cas, de déterminer des solutions pour résoudre ces conflits. L'objectif est de parvenir à une application déterministe des règles (en un passe et sans retour arrière), cela afin de limiter la complexité algorithmique du module de prétraitement (et par conséquent celle du système SYGET).

En étudiant de près la nature des règles et la façon dont elles doivent être appliquées sur le corpus, nous relevons une difficulté quand, dans l'ontologie d'extraction, un concept C_Y est défini par une combinaison de concepts dont un au moins permet de définir un autre concept C_Z. Cette difficulté se traduit au niveau des règles par les deux problèmes suivants :

1. Une règle disjonctive <C_Y> ::= { <C_A> ∨ <C_B> } ;; est transformée en une règle `RSx C_A | C_B -> C_Y` et une règle `RDz C_A |+ C_B -> C_Y`. Si la règle `RSx` est appliquée avant la règle `RDz`, alors chaque combinaison d'une instance de C_A et d'une instance de C_B dans le texte définira deux instances de C_Y et non pas une unique instance de C_Y. Une telle application produirait ainsi un résultat non conforme avec l'ontologie d'extraction. Pour éviter ce type de problème il est nécessaire de s'assurer que la règle `RDz` issue d'une règle disjonctive sera exécutée avant la règle `RSx` issue de cette même règle disjonctive ;

2. Lorsqu'un concept C_A présent dans la prémisse d'une règle RC intervient dans la prémisse d'une règle RS (issue de la transformation d'une règle sélective ou d'une règle disjonctive), l'ordre d'application des règles peut provoquer des erreurs d'identification des instances de concepts dans les textes. Prenons par exemple deux règles `RCx C_A + C_B -> C_E` et `RSz C_A1 | C_A| C_A2 -> C_D`. Si la règle `RSz` est appliquée avant la règle `RCx`, chaque instance du concept C_A trouvée dans le texte sera marquée et reconnue comme une instance du concept C_D. Aussi chaque suite d'une instance de C_A et d'une instance de C_B dans le texte sera identifiée par le système comme la suite d'une instance de C_D et d'une instance de C_B. En conséquence la règle `RCx` ne pourra pas être appliquée et les instances du concept C_E présentes dans le texte ne seront pas identifiées. Pour s'abstraire de ce problème, il apparaît nécessaire que la partie de la règle `RSz` traitant C_A (qui peut être réécrite en une règle `C_A -> C_D`) soit appliquée après la règle `RCx`.

Afin de résoudre ces problèmes le module de génération de la base de règles procède à un réordonnancement des règles en découpant le contenu du fichier « règles constitutives concept » en quatre fichiers :

- « règles sélectives 1 » : ce fichier contient les règles RS qui n'entrent en conflit avec aucune règle RD ou RC, il s'agit des règles qui doivent être appliquées en premier ;

- « règles disjonctives » : il contient les règles RD ;

- « règles conjonctives » : il contient les règles RC ;
- « règles sélectives 2 » : ce fichier est constitué des règles RS qui entrent en conflit avec les règles RD et RC. Elles doivent être appliquées après toutes les autres.

Pour déterminer quelles sont les règles RS qui doivent être placées dans le fichier <règles sélectives 2>, nous utilisons la méthode suivante (formalisée par l'algorithme de la figure 7.4). Toutes les règles RS sont d'abord copiées dans le fichier « règles sélectives 1 ». Ensuite le fichier <règles constitutives concept> est analysé en regardant, pour chaque concept C_A_i présent dans la prémisse d'une règle RS, s'il n'est pas également présent dans la prémisse d'une règle RSx C_A_1|C_A_2|...|C_A_i|...|C_A_n -> C_Y. Si c'est le cas la règle RSx est remplacée dans le fichier « règles sélectives 1 » par la règle RSx C_A_1|C_A_2|...|C_A_{i-1}|C_A_{i+1}|...|C_A_n -> C_Y puis une nouvelle règle RSz C_A_i -> C_Y est créée et enregistrée dans le fichier « règles sélectives 2 ».

```
Algorithme ordonnancement_règles

Début
   Pour toute règle R du fichier règles constitutives concept faire
        Si type_règle(R) = RS alors
            ajoute_règle(R, règles sélectives 1)
        Finsi
   Finpour

   Pour toute règle R1 du fichier règles constitutives concepts faire
     Si (type_règle(R1) = RC) ou (type_règle(R1) = RD) alors
        Pour tout concept C_A_i de partie_gauche(R1) faire
            Pour toute règle R2 de règles constitutives concept faire
                Si est_en_partie_gauche(C_A_i, R2) alors
                    enlève_règle(R2, règles sélectives 1)
                    // la fonction supprime_concept_partie_gauche(C_A_i,R2)
                    // transforme la règle RSx C_A_1|C_A_2|...|C_A_i|...|C_A_n -> C_D
                    // en RSx C_A_1|C_A_2|...|C_A_i-1|C_A_i+1|...|C_A_n -> C_D
                    R3 <- supprime_concept_partie_gauche(C_A_i, R2)
                    ajoute_règle(R3, règles sélectives 1)
                    // la fonction création_règle_RS(arg1, arg2) crée une règle
                    // de type RS arg1 -> arg2
                    R4 <- création_règle_RS(C, partie_droite(R2))
                    ajoute_règle(R4, règles sélectives 2)
                Fin Si
            Fin Pour
        Fin Pour
     Fin Si
  Fin Pour

Fin ordonnancement_règles
```

Figure 7.4 : Algorithme d'ordonnancement des règles constitutives sur les concepts

La méthode précédente permet de s'abstraire des conflits posés par les règles RS mais un problème subsiste néanmoins concernant l'ordre d'application des règles RD et des règles RC. Le système doit-il appliquer les règles RD avant ou après les règles RC ? Des conflits entre règles RD et RC sont possibles car **MEGET** autorise qu'un concept appartenant à l'ensemble de définition d'une règle disjonctive appartienne également à l'ensemble de définition d'une règle conjonctive. Dans le cas de deux règles `RDx C_A|+C_B -> C_Y` et `RCy C_B+C_E -> C_D`, ces deux règles entrent en conflit si une suite d'instances des concepts C_A, C_B et C_E est trouvée dans le texte. Si la règle `RDx` est appliquée d'abord, le système identifiera cette suite comme une instance de C_Y suivie d'une instance de C_E. La règle `RCy` ne peut alors plus être appliquée. À l'inverse, si la règle `RCy` est appliquée d'abord, le système identifiera la suite comme une instance de C_A suivie d'une instance de C_D et la règle `RDx` n'est plus applicable. De tels cas sont néanmoins très rares (nous n'en avons relevé aucun lors de nos expérimentations sur le corpus [CREC]).

Pour illustrer un tel cas de figure prenons comme exemple un corpus traitant de droit et une ontologie définissant le concept de loi par un descripteur de loi ("*loi*", "*arrêt*", etc.) mais aussi par un descripteur et un nom ("*loi Evin*", "*loi De Robien*"), et définissant le concept de personne par un nom, un prénom ou les deux. Les règles formelles décrivant ces concepts donnent, après transformation, les règles suivantes :

```
RS1 C_DC_LOI -> C_LOI
RS2 C_NOM | C_PRENOM->C_PERSONNE
RC1 C_DC_LOI + C_NOM -> C_LOI
RD1 C_NOM |+ C_PRENOM -> C_PERSONNE
```

Ici un conflit se produit entre la règle `RD1` et la règle `RC1` à chaque fois qu'est trouvée dans le texte une suite d'instances de C_DC_LOI, C_NOM et C_PRENOM. Si `RD1` est appliquée d'abord, le système identifie cette suite comme une instance de C_DC_LOI suivie d'une instance de C_PERSONNE. À l'inverse si `RC1` est appliquée en premier, alors le système identifie la suite comme une instance de C_LOI suivie d'une instance de C_PRENOM.

Pour résoudre ces problèmes, nous avons choisi arbitrairement de favoriser l'application des règles RD avant les règles RC. Ce choix est fondé sur l'hypothèse que lorsqu'un rédacteur place côte à côte plusieurs termes correspondant à des concepts présents dans l'ensemble de définition d'une règle disjonctive Rg, il désire prioritairement exprimer la notion exprimée par le concept décrit par Rg. Ainsi par défaut **SYGET** est paramétré pour exécuter d'abord les règles contenues dans le fichier `« règles disjonctives »` et ensuite celles contenues dans le fichier `« règles conjonctives »`.

Notons que l'hypothèse précédente est dictée par la langue du corpus et n'est pas forcement vraie pour toutes les langues. Le système peut facilement être adapté au traitement d'un corpus écrit dans une langue dans laquelle cette hypothèse ne se vérifie pas. Il suffit alors de le paramétrer pour qu'il exécute les règles contenues dans `« règles conjonctives »` avant celles contenues dans `« règles disjonctives »`.

7.2.2.3 Paramétrage des règles

Après la création des différents fichiers issus des règles constitutives sur les concepts, le module propose un paramétrage des règles du type RC et RD. Ce paramétrage consiste à fixer la valeur d'un paramètre dit de *séparation* pour chaque règle. Cette possibilité est donnée à l'utilisateur afin de lui permettre de mieux adapter le système à des particularités du corpus à traiter, et ainsi d'améliorer les performances du processus d'extraction. Le paramétrage peut être réalisé entièrement *a priori* mais de meilleurs résultats sont obtenus lorsqu'il est le résultat de quelques expérimentations successives sur une petite portion du corpus à traiter.

En appliquant une règle `RCx C_A+C_B -> C_`Y, le système identifie une instance de C_Y par la présence d'une instance de C_A suivie par une instance de C_B (dans cet ordre). Avec une règle `RDx C_A|+C_B -> C_Y`, il identifie une instance de C_Y par la présence d'une instance de C_A suivie ou précédée par une instance de C_B. Par défaut une instance de C_Y est identifiée si les instances de C_A et de C_B sont séparées par un ou plusieurs espaces, un mot vide[1] ou par certains signes de ponctuation (virgule, point-virgule, deux points). Il s'agit des éléments intercalaires par défaut. Tout autre élément séparant ces deux instances dans le texte empêche le système de les identifier comme une instance de C_Y.

Le paramètre de séparation permet de définir d'autres éléments pouvant s'intercaler entre des instances de concepts présents dans la prémisse d'une règle de type RC ou RD. Ces éléments intercalaires peuvent être des caractères, des mots, des termes ou bien des concepts. Le paramètre permet également de fixer un nombre d'occurrences pour chacun des éléments intercalaires (1 fois, n fois ou de n à m fois) ; lorsque le nombre d'occurrences n'est pas fixé, l'élément intercalaire peut apparaître de 1 à une infinité de fois. Une valeur nulle pour le paramètre de séparation signifie au système qu'il doit utiliser les éléments intercalaires par défaut.

Une fois les valeurs du paramètre fixées pour une règle, le module modifie cette règle dans la base en ajoutant après son identifiant (catégorie et numéro de la règle) un ensemble d'unités lexicales décrivant les différents éléments intercalaires ainsi que leur nombre d'occurrences.

[1] Plusieurs définitions des mots vides [Vergne 2003] [Houben 2004] coexistent. Dans notre cas nous considérons comme mots vides l'ensemble des mots-outils tels qu'ils ont été cités dans le chapitre 2 (cf. note n°18, page 46).

7.2.3 Règles prédicatives

Le module de génération de la base de règles transforme automatiquement les règles prédicatives et les place dans un fichier `<règles prédicatives>`.

La règle prédicative (où m, t, n_i et i $\in \mathbb{N}$)

<C_Y> ::= {
descripteur = <C_DC_Y> ;
objet = <C_OBJ$_1$> | <C_OBJ$_2$> | ... | <C_OBJ$_m$> ;
option_1 = <C_O1$_1$> | <C_O1$_2$> | ... | <C_O1$_{n1}$> ;
option_2 = <C_O2$_1$> | <C_O2$_2$> | ... | <C_O2$_{n2}$> ;
... ;
option_t = <C_Ot$_1$> | <C_Ot$_2$> | ... | <C_Ot$_{nt}$> .
} ;;

devient après transformation, l'expression suivante :

```
C_Y   :   C_DC_Y
          C_Y_objet     : C_OBJ1 | C_OBJ2 |  ... | C_OBJm
          C_Y_option_1  : C_O11 | C_O12 | ... | C_O1n1
          C_Y_option_2  : C_O21 | C_O22 | ... | C_O2n2
          ...
          C_Y_option_t  : C_Ot1 | C_Ot2 | ... | C_Otnt
```

Lorsque C_Y est décrit par une règle prédicative dans laquelle une (ou plusieurs) option est définie par :

option_q == < C_Oq$_1$> | < C_Oq$_2$> | ... | < C_Oq$_{nq}$> (cf. section 5.3.2.4)

cette option est traduite par l'expression

`C_Y_option_q :: C_Oq1 | C_Oq2 | ... | C_Oqnq .`

Exemple 7.14 (Transformation d'une règle prédicative)

La règle suivante décrit le concept C_ACHAT

```
<C_ACHAT>  ::=  {
                    descripteur  =  <C_DC_ACHAT> ;
                    objet        =  <C_IMMOBILIER> | <C_VEHICULE> |
                                    <C_PRODUIT_BANCAIRE> ;
                    datation     =  <C_DATE> ;
                    montant      =  <C_SOMME> ;
                    localisation =  <C_LIEU> .
                };;
```

Elle devient après transformation :

```
C_ACHAT : C_DC_ACHAT
          C_ACHAT_objet : C_IMMOBILIER | C_VEHICULE | C_PROD_BANCASS
          C_ACHAT_date : C_DATE
          C_ACHAT_localisation : C_LIEU
          C_ACHAT_montant : C_SOMME
```

7.3 Module d'étiquetage

Ce module procède à un étiquetage du texte qui marque chaque instance d'un concept de l'ontologie par des balises XML. Le système étiquette le corpus en appliquant les règles contenues dans les fichiers créés par le module de génération de la base de règles. L'étiquetage se déroule en deux étapes [Even 2004] : un premier étiquetage utilise les règles constitutives (module d'étiquetage constitutif) et un deuxième applique les règles informatives et prédicatives (module d'étiquetage prédicatif). Le résultat de l'exécution du module d'étiquetage (cf. figure 7.5) est un fichier XML dans lequel les concepts et relations décrits dans l'ontologie apparaissent clairement grâce aux balises (cf. exemple 7.18 et figure 7.7).

7.3.1 Étiquetage constitutif

L'étiquetage constitutif est effectué au moyen de trois sous-modules successifs : un premier traite les concepts génériques, un deuxième applique les règles sélectives terminales (étiquetage des termes) et un troisième applique les règles de type RS, RC et RD (étiquetage des concepts). L'étiquetage est réalisé en parcourant le corpus de gauche à droite.

Lors de la phase d'étiquetage constitutif, les concepts sont toujours repérés dans le texte par les balises les plus extérieures. Les balises intérieures lorsqu'elles existent, ne sont pas utilisées pour identifier les concepts.

Par exemple `<C_IMMOBILIER><C_APPARTEMENT>t2</C_APPARTEMENT></C_IMMOBILIER>` est identifié comme une instance du concept C_IMMOBILIER.

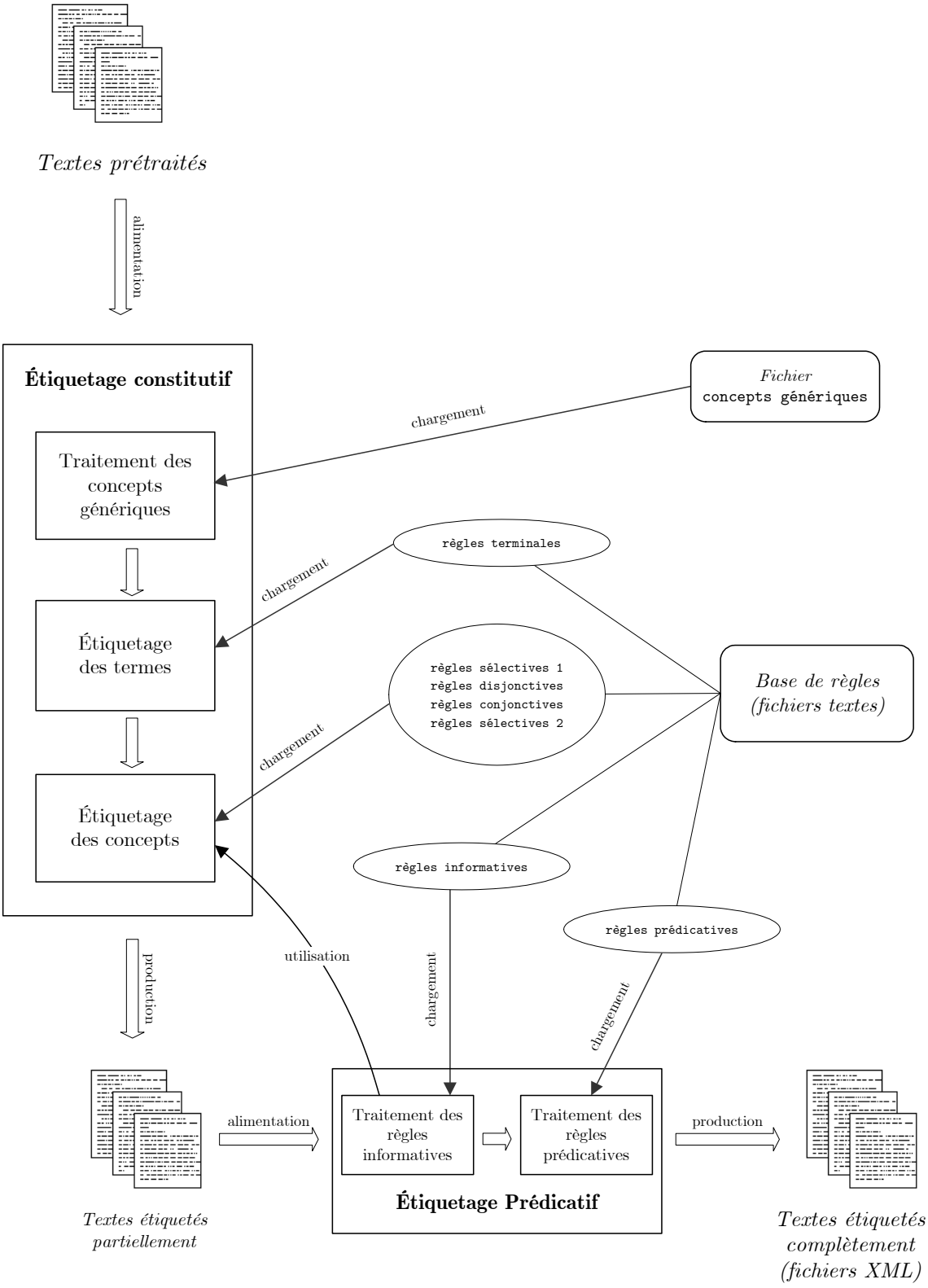

Figure 7.5 : Module d'étiquetage

7.3.1.1 Traitement des concepts génériques

Les descriptions des concepts génériques (descriptions spécifiées lors de l'élaboration de l'ontologie des besoins et enrichies lors de l'étude termino-ontologique) sont transcrites manuellement en un ensemble de règles contextuelles de réécriture qui sont placées dans le fichier `<concepts génériques>`. Ces règles sont souvent étroitement liées à la langue de rédaction des textes. Le sous-module traitant les concepts génériques procède à l'étiquetage en balisant chaque production lexicale d'un concept générique (décrite dans la partie gauche d'une règle de réécriture), par l'identifiant de ce concept (cf. exemple 7.15). Cet étiquetage est facilité par la phase de prétraitements qui transforme en un format unique les dates au format numérique ainsi que les productions des concepts du type mesure.

Exemple 7.15 (Étiquetage des concepts génériques)

Le texte de l'exemple 7.1, issu de l'extrait de corpus original

projet achat 04/2003 bmw serie 3 30ke

devient, après l'exécution du sous-module d'étiquetage des concepts génériques, le texte suivant :

projet achat **<C_DATE><C_DATE_FLOUE_1>***04/2003***</C_DATE_FLOUE_1></C_DATE>**
bmw serie 3 **<C_SOMME>***30ke***</C_SOMME>**

Les règles écrites pour les concepts génériques s'avèrent générales à la plupart des textes à condition toutefois qu'ils soient traités au préalable par le module de prétraitement. Si des modifications sont nécessaires (adaptation à une nouvelle langue par exemple), le module n'est pas modifié, les changements sont réalisés uniquement au niveau des règles. Les modifications sur les règles sont facilitées par l'utilisation du formalisme de réécriture.

7.3.1.2 Étiquetage des termes

Ce sous-module repère dans le texte toutes les instances des concepts de base et des concepts descripteurs (de prédicats et de concepts) de l'ontologie. Ces concepts sont décrits par des règles du fichier `<règles terminales>`.

Le fichier `<règles terminales>` est chargé par le système. Ensuite chacune des règles `' (`t_1 `|` t_2 `|` t_3`| ... |` t_n`) ' -> ' <C_Y>$1</C_Y> '` présente dans le fichier est appliquée par le système en remplaçant chaque terme t_1 ou t_2 ou ... ou t_n trouvé dans le texte par la séquence lexicale formé par ce terme (mémorisée dans la variable `$1`) balisé par l'identifiant du concept qu'il lexicalise (concept C_Y).

Lorsque toutes les règles sont appliquées, toutes les occurrences dans le texte des termes présents dans l'ensemble de règles formelles représentant l'ontologie d'extraction (soit tous les termes du corpus jugés pertinents pour l'extraction d'informations) sont étiquetés.

Exemple 7.16 (Étiquetage des termes)

L'extrait de corpus de l'exemple 7.15 devient, après l'exécution du sous-module d'étiquetage des termes, le texte suivant :

<C_DC_PROJET>*projet*<C_DC_PROJET> <C_DC_ACHAT>*achat*</C_DC_ACHAT>
<C_DATE><C_DATE_FLOUE_1>*04/2003*</C_DATE_FLOUE_1></C_DATE>
<C_MARQUE_AUTO>*bmw*</C_MARQUE_AUTO> <C_MODELE_AUTO>*serie 3*</C_MODELE_AUTO>
<C_SOMME>*30ke*</C_SOMME>

par l'application des règles de réécriture :

```
' achat | rachat | acheter ' -> ' <C_DC_ACHAT>$1</ C_DC_ACHAT> '
' projet | projete de ' -> ' < C_DC_PROJET>$1</ C_DC_PROJET> '
' bmw | citroen | renault | mercedes ' -> ' < C_MARQUE_AUTO>$1</ C_MARQUE_AUTO> '
' serie 3 | cx | testa rossa | twingo ' -> ' < C_MODELE_AUTO>$1< C_MODELE_AUTO> '
```

qui correspondent respectivement aux règles formelles suivantes :

< C_ DC_ ACHAT> ::= { achat / rachat / acheter } ;;
< C_ DC_ PROJET> ::= { projet / projete de} ;;
< C_ MARQUE_ AUTO ::= { bmw / citroen / renault / mercedes } ;;
< C_ MODELE_ AUTO> ::= {serie 3 / cx / testa rossa / f40 / twingo } ;;

7.3.1.3 Étiquetage des concepts

Le sous-module d'étiquetage des concepts applique les règles présentes dans les fichiers `« règles sélectives 1 »`, `« règles disjonctives »`, `« règles conjonctives »` et `« règles sélectives 2 »`.

Il est composé de quatre processus correspondant chacun à un fichier et est paramétré pour les exécuter successivement dans l'ordre établi pour résoudre les problèmes de conflits entre règles (cf. section 7.2.2.2) :

1. Processus d'application des règles RS 1 (fichier `« règles sélectives 1 »`) ;
2. Processus d'application des règles RD (fichier `« règles disjonctives »`) ;
3. Processus d'application des règles RC (fichier `« règles conjonctives »`) ;
4. Processus d'application des règles RS 2 (fichier `« règles sélectives 2 »`).

Un processus se termine lorsque plus aucune des règles n'est applicable. Lorsque le dernier processus est terminé, le sous-module effectue une nouvelle passe (exécution des processus 1 à 4) et ainsi de suite. Le module s'arrête quand aucune règle n'a été appliquée lors d'une passe, c'est-à-dire lorsque que le texte reste inchangé après le passage à travers les quatre processus.

L'exemple 7.17 à la fin de cette section présente le résultat de l'exécution du sous-module d'étiquetage des concepts sur le texte de l'exemple 7.16.

7.3.1.3.1 Processus d'application des règles RS

Les deux processus d'application des règles RS utilisent le même principe. Pour chaque règle `C_A1 | C_A2 | ... | C_An -> C_Y`, le système repère un concept C_A_i dans le texte grâce à ses balises et rajoute les balises de C_Y autour de celles de C_A_i.

7.3.1.3.2 Processus d'application des règles RC

Pour chaque règle `RSx C_A1 + C_A2 +...+ C_An -> C_Y`, le processus d'application des règles conjonctives cherche dans le texte les instances des concepts C_A_i. Dans les cas où une instance de C_A_1, une instance de C_A_2, ... et une instance de C_A_n sont trouvées côte à côte et dans cet ordre, le processus encadre les n instances des concepts par les balises de C_Y.

Des instances de concepts sont considérées côte à côte si elles sont séparées par les éléments intercalaires définis par le paramètre de séparation de la règle (cf. section 7.2.2.3) ou par les éléments intercalaires par défaut dans le cas ou ce paramètre n'a pas été valué.

7.3.1.3.3 Processus d'application des règles RD

Le processus d'application des règles disjonctives génère la combinatoire pour chaque règle `C_A1 |+ C_A2 |+ ... |+ C_An -> C_Y`, c'est-à-dire produit à partir d'une telle règle, un nouvel ensemble de règles dont la partie droite est C_Y et la partie gauche est une des combinaisons possibles des concepts C_A_i. Toutes les combinaisons possibles sont traitées.

Par exemple à partir de la règle `C_A1 |+ C_A2 |+ C_A3 -> C_Y`, le processus génère l'ensemble de règles suivant :

```
C_A1 + C_A2 + C_A3 -> C_Y ; C_A1 + C_A3 + C_A2 -> C_Y ; C_A2 + C_A1 + C_A3 -> C_Y ;
C_A2 + C_A3 + C_A1 -> C_Y ; C_A3 + C_A2 + C_A1 -> C_Y ; C_A3 + C_A1 + C_A2 -> C_Y ;
C_A1 + C_A2 -> C_Y ; C_A1 + C_A3 -> C_Y ;
C_A2 + C_A3 -> C_Y ; C_A3 + C_A2 -> C_Y
```

Chacune des règles produites est exécutée de la même manière qu'une règle conjonctive.

Exemple 7.17 (Étiquetage des concepts)

L'extrait de corpus de l'exemple 7.16 devient, après l'exécution du sous-module d'étiquetage des concepts, le texte suivant :

```
<C_DC_PROJET>projet<C_DC_PROJET> <C_DC_ACHAT>achat</C_DC_ACHAT>
<C_DATE><C_DATE_FLOUE_1>04/2003</C_DATE_FLOUE_1></C_DATE>
<C_VEHICULE><C_AUTO>
  <C_MARQUE_AUTO>bmw</C_MARQUE_AUTO>
  <C_MODELE_AUTO>serie 3</C_MODELE_AUTO>
</C_AUTO></C_VEHICULE>
<C_SOMME>30ke</C_SOMME>
```

en utilisant les règles formelles :

< C_AUTO> ::= { < C_MARQUE_AUTO> ∨ < C_MODELE_AUTO> } ;;
< C_VEHICULE> ::= { < C_AUTO> / < C_MOTO> / <C_DIV_VEHICULE> } ;;

transformées par le module de génération de la base de règles en l'ensemble de règles suivant :

```
C_AUTO | C_MOTO | C_DIV_VEHICULE -> C_VEHICULE
C_MARQUE_AUTO | C_MODELE_AUTO -> C_AUTO
C_MARQUE_AUTO |+ C_MODELE_AUTO -> C_AUTO
```

7.3.2 Étiquetage prédicatif

L'étiquetage prédicatif repère les instances des concepts décrits par des règles mettant en jeu des arguments, c'est-à-dire les règles informatives et prédicatives du système. Le module procède à l'application successive de ces deux types de règles en commençant par les règles informatives.

7.3.2.1 Traitement des règles informatives

7.3.2.1.1 Fonctionnement du processus

Pour chaque instance d'un concept dont l'identifiant commence par « C_DF_ », le processus applique la règle informative qui lui est associée dans le fichier `<règles informatives>`.

Une règle informative « `C_Y : C_DF_Y C_Y_attribut_1 : C_A1`$_1$ `| ... | C_A1`$_{n1}$ `... C_Y_attribut_t : C_At`$_1$ `| ... | C_At`$_{n1}$ » est appliquée de la façon suivante :

1. L'instance de C_DF_Y est balisé par l'identifiant du concept C_Y et un numéro d'instance (`<C_Y instance= "num">...</C_Y>`) ;
2. Le système cherche à valuer les différents attributs de C_Y pour cette instance. Pour chaque attribut *at*, il recherche la première instance d'un des concepts déterminés comme type de *at*. (par exemple pour `attribut_1`, il recherchera la première instance d'un des concepts C_A1$_1$, ..., C_A1$_{n1}$). S'il en trouve une, il la balise par l'identifiant de C_Y, le numéro de l'instance qu'il est en train de traiter et le nom de l'attribut (`<C_Y instance="num" arg="at">...</C_Y>`). Dans le cas contraire, l'attribut ne sera pas marqué dans le texte ;
3. L'application de la règle se termine une fois que tous les attributs sont traités (qu'une valeur ait été trouvée ou non).

Quand toutes les instances des concepts « C_DF_X_i » ont été traitées, le système procède à une exécution du processus d'étiquetage des concepts (cf. section 7.3.1.3). En effet l'application des règles informatives peut faire apparaître dans le texte de nouvelles instances de concepts présents en prémisse de règles RS, RD ou RC, instances non traitées auparavant par le système.

L'ensemble de la procédure est réitéré jusqu'à ce que plus aucune règle informative ne soit applicable, c'est-à-dire lorsque après une itération de la procédure, le texte n'est pas modifié (aucune balise n'est ajoutée lors de l'itération).

L'exécution du processus sur un extrait de corpus est illustrée par l'exemple 7.18.

Exemple 7.18 (Traitement des règles informatives)

À partir du texte

> *proposition pret immo de 75kf sur 24mois 5,70%*

le module d'étiquetage constitutif produit le texte suivant (en appliquant notamment la règle `C_PRET_AUTO | C_PRET_IMMO | C_PRET_CONSO -> C_DF_PRET)` *:*

```
<DC_PROPOSITION>proposition</DC_PROPOSITION>
<C_DF_PRET>
    <C_PRET_IMMO>
        <DC_PRET>pret</DC_PRET>
        <C_IMMOBILIER><C_DC_IMMOBILIER>immo</C_DC_IMMOBILIER></C_IMMOBILIER>
    </C_PRET_IMMO>
</C_DF_PRET>
<C_SOMME>75ke</C_SOMME> sur <C_DUREE>24mois</C_DUREE> <C_TAUX>5,70%</C_TAUX>
```

Après exécution du processus de traitement des règles informative, le texte précédent devient :

```
<DC_PROPOSITION>proposition</DC_PROPOSITION>
<C_PRET instance="1">
    <C_DF_PRET>
      <C_PRET_IMMO>
        <DC_PRET>pret</DC_PRET>
        <C_IMMOBILIER><C_DC_IMMOBILIER>immo</C_DC_IMMOBILIER></C_IMMOBILIER>
      </C_PRET_IMMO>
    </C_DF_PRET>
</C_PRET>
<C_PRET instance="1" arg="montant">
    <C_SOMME>75ke</C_SOMME>
</C_PRET>
sur
<C_PRET instance="1" arg="durée">
    <C_DUREE>24mois</C_DUREE>
</C_PRET>
<C_PRET instance="1" arg="taux">
    <C_TAUX>5,70%</C_TAUX>
</C_PRET>
```

Cet étiquetage correspond à une application de la règle suivante :

```
C_PRET :    C_DF_PRET
            C_PRET_durée   : C_DUREE
            C_PRET_taux    : C_TAUX
            C_PRET_montant : C_SOMME
```

7.3.2.1.2 Paramètres de recherche des valeurs d'attributs

Les instances des types d'attributs d'un concept C_Y sont toujours recherchées dans les textes à partir de l'instance du concept C_DF_Y. Par défaut, cette recherche est effectuée de gauche à droite dans une fenêtre de recherche bornée à gauche par l'instance de C_DF_Y et à droite par : soit une instance d'un concept C_DF_X_i d'une règle informative, soit une instance d'un descripteur de prédicat, soit un marqueur de fin de phrase (signe de ponctuation forte, suite de tabulations ou d'espace, retour à la ligne, etc.).

Pour chaque attribut, la taille de la fenêtre ainsi que la direction de la recherche peuvent être paramétrés :

- *Direction* : ce paramètre permet de choisir un sens de recherche parmi quatre possibilités :
  1. De gauche à droite : il s'agit de la valeur par défaut ;
  2. De droite à gauche : dans ce cas la fenêtre de recherche est alors bornée à droite par l'instance de C_DF_Y ;
  3. De gauche à droite puis de droite à gauche : une recherche de gauche à droite est d'abord réalisée. Si aucun résultat n'est trouvé, le système effectue ensuite une recherche dans le sens inverse ;
  4. De droite à gauche puis de gauche à droite : si la recherche de droite à gauche échoue, le système effectue une recherche dans le sens inverse.
- *Taille de la fenêtre* : la taille de la fenêtre peut être fixée en nombre de termes ou en nombre d'instance de concepts. Ce paramètre permet surtout de limiter la distance de recherche afin d'éviter les valuations erronées d'arguments (il est peu probable que deux instances de concept éloignées soient en relation).

À l'instar des paramètres des règles RS et RD, les valeurs de ces paramètres peuvent être définies *a priori* mais de meilleurs résultats sont obtenus si ceux-ci sont fixés après quelques expérimentations sur un extrait du corpus à traiter.

7.3.2.2 Traitement des règles prédicatives

7.3.2.1.1 Fonctionnement du processus

À chaque fois que le processus trouve une instance d'un des concepts définis comme descripteur de prédicat, il applique la règle associée à ce concept et définie dans le fichier `« règles predicatives »` (cf figure 7.6). L'application se déroule différemment selon que la règle est définie avec ou sans l'argument objet.

- La règle décrivant un concept C_Y est définie avec un objet (`C_Y : C_DC_Y C_Y_objet : C_OBJ`$_1$ `| ... |` `C_OBJ`$_{n1}$ `C_Y_options_1 : C_O1`$_1$ `| ... |` `C_O1`$_{n1}$ `... C_Y_option_t : C_Ot`$_1$ `| ... |` `C_Ot`$_{nt}$). Le processus commence par rechercher la plus proche instance d'un des concepts C_OBJ$_i$ définis comme type d'objet. Par défaut la recherche de l'objet est réalisée de gauche à droite dans une fenêtre bornée à gauche par l'instance du descripteur (instance de C_DC_Y) et à droite par le prochain concept descripteur d'un prédicat (ce concept est inclus dans l'intervalle de recherche) ou par un marqueur de fin de phrase (signe de ponctuation forte, suite de tabulations ou d'espace, retour à la ligne, etc.). Mais les paramètres de recherche peuvent être modifiés comme pour les règles informatives (cf. section 7.3.2.2.2).

  La présence d'une instance d'un concept C_OBJ$_i$ marque l'identification d'une instance I$_{C_Y}$ de C_Y et l'instance de C_OBJ$_i$ value l'objet pour I$_{C_Y}$. Le processus balise l'instance de C_DC_Y par l'identifiant de C_Y et un numéro d'instance (pour gérer les cas ou plusieurs instances de C_Y sont présentes dans le texte) et l'instance de C_OBJ$_i$ par l'identifiant de C_Y, le numéro d'instance de I$_{C_Y}$ et le terme *objet* (`<C_Y instance="num" arg="objet"> ...</C_Y>`). À l'inverse si aucun concept défini comme type d'objet n'est trouvé pour cette instance du descripteur C_DC_Y, elle est ignorée (le système n'identifie pas d'instance de C_Y).

- Lorsque la règle décrivant un concept C_Y est définie sans objet (`C_Y : C_DC_Y C_Y_options_1 : C_O1`$_1$ `| ... |` `C_O1`$_{n1}$ `... C_Y_option_t : C_Ot`$_1$ `| ... |` `C_Ot`$_{n1}$), une instance de C_Y est identifiée pour chaque instance I$_{DESC}$ de C_DC_Y. I$_{DESC}$ est alors balisée par l'identifiant de C_Y et un numéro d'instance (`<C_Y instance="num">...</C_Y>`).

Quelle que soit la règle prédicative décrivant le concept C_Y (avec ou sans objet), lorsqu'une instance de C_Y est identifiée, le système cherche à valuer les options définies dans la règle. Lors de cette phase (détaillée dans la section suivante) toutes les options ne sont pas forcément valuées car les informations ne sont pas obligatoirement présentes dans le corpus.

L'ensemble de la procédure précédente est répété jusqu'à ce que plus aucune règle prédicative ne soit applicable, c'est-à-dire lorsque après une itération de la procédure, le texte n'est pas modifié (aucune balise n'est ajoutée lors de l'itération). À ce point, les instances de descripteurs non traités (c'est à dire correspondant à des règles dont l'argument objet n'a pu être valué) ne pourront l'être. Dans ce cas ils sont marqués comme décrivant des instances vides de concept (balisées par l'identifiant du concept et avec la valeur *NULL* comme numéro d'instance, cf. exemple 7.19).

Exemple 7.19 (Instance vide de concept)

L'extrait de corpus « *mr deniaud a un projet* » dans lequel l'objet du projet n'est pas précisé, devient après étiquetage le texte suivant :

```
<C_PERSONNE>
    <C_ID_NOM>mr</C_ID_NOM> <C_NOM>deniaud</C_NOM>
</C_PERSONNE>
a un
<C_PROJET instance=NULL>
    <C_DC_PROJET>projet</C_DC_PROJET>
</C_PROJET>
```

7.3.2.1.2 Recherche des valeurs d'options

Le processus qui cherche à valuer les options pour une instance I_{C_A} d'un concept C_A décrit par une règle prédicative Rg_{C_A}, gère trois cas :

1. La règle Rg_{C_A} est définie sans l'argument objet ;
2. La règle Rg_{C_A} est définie avec l'argument objet et la valeur de l'objet pour I_{C_A} est une instance I_{C_B} d'un concept C_B qui n'est pas décrit par une règle informative ou prédicative ;
3. La règle Rg_{C_A} est définie avec l'argument objet et la valeur de l'objet pour I_{C_A} est une instance I_{C_B} d'un concept C_B qui est décrit par une règle informative ou prédicative Rg_{C_B}.

Dans les deux premiers cas, la recherche des valeurs des options pour I_{C_A} est effectuée en procédant d'une manière similaire à celle de la recherche de l'objet. Pour chaque option opA_i, le système recherche la première instance d'un des concepts déterminés comme type de l'option. Lorsqu'une instance de concept est trouvée, cette instance est balisée par une expression formée de l'identifiant de C_A, de son numéro d'instance et du nom de l'option (`<C_A instance="num" arg="opA`$_i$`">...</C_A>`).

Dans le troisième cas, le processus prend en compte la possibilité qu'une ou plusieurs options de Rg_{C_A} soient considérées identiques (cf. section, 5.3.2.4) à des options de la règle Rg_{C_B}.

Options différentes

Si une option opA_i de Rg_{C_A} est différente d'une option (ou un attribut) opB_j de Rg_{C_B}, cet aspect est exprimé dans Rg_{C_A} par la présence du symbole « : » entre l'identifiant de l'option (formé de l'identifiant du concept décrit par Rg_{C_A} et du nom de l'option, ici `C_A_opA`$_i$) et la liste de ses types :

$$\texttt{C\_A\_opA}_i \texttt{ : C\_O}_1\texttt{|C\_O}_2\texttt{|...|C\_O}_n$$

Lorsqu'une option opA_i de Rg_{C_A} est exprimée avec le symbole « : », le système la value pour I_{C_A} indépendamment de la valeur de l'objet en procédant de la même manière que celle présentée pour les cas 1 et 2 ci-dessus.

Options identiques

Dans la base de règles, lorsqu'une une option opA_i de Rg_{C_A} est considérée identique à une option (ou un attribut) opB_j de Rg_{C_B},, cet aspect est exprimé dans Rg_{C_A} par la présence du symbole « :: » entre l'identifiant de l'option opA_i (ici `C_A_opA`$_i$) et la liste des types d'opA_i :

`C_A_opA`$_i$ `:: C_O`$_1$`|C_O`$_2$`|...|C_O`$_n$

La méthode de construction de l'ontologie d'extraction impose que opA_i et opB_j aient le même nom lorsque $opA_i == opB_j$. Ainsi dans Rg_{C_B}, opB_j est nommée par un identifiant formé de « `C_B` » et du nom de l'option opA_i (`C_B_opA`$_i$).

Le module d'étiquetage prédicatif gère le cas des options identiques de la façon suivante : lorsque le module trouve une instance I_{C_A} d'un concept C_A décrit par une règle prédicative dans laquelle une option `C_A_opA`$_i$ est définie avec le symbole « `::` », il regarde si la valeur de l'objet pour I_{C_A} est une instance I_{C_B} d'un concept C_B décrit par une règle possédant une option `C_B_opA`$_i$. Dans ce cas, le système value l'option opA_i pour I_{C_A} par la valeur de opA_i pour I_{C_B}, si elle existe. Concrètement, le système identifie la valeur de opA_i pour I_{C_B} grâce à ses balises et l'étiquette par la référence de I_{C_A} (identifiant de C_A et numéro d'instance de I_{C_A}) et le nom de opA_i (`<C_A instance="num I`$_{C\_A}$`" arg="opA`$_i$`"> ... </C_A>`).

A contrario, si opB_j n'est pas valuée pour I_{C_B} (aucune instance de concept n'a été trouvée pour cette option), l'option opA_i de I_{C_A} ne sera volontairement pas valuée par le système.

```
Algorithme étiquetage_prédicatif

Début
    charge(<règle prédicatives>)
    // initialisation des numéros d'instance pour tous les prédicats
    Pour tous les prédicats P
        num_instance[P]=0
    Fin Pour
    est_applicable = VRAI
    // application des règles prédicatives tant qu'il
    // est possible de le faire
    Tant Que est_applicable faire
        TEXTE_INIT ← TEXTE
        // recherche du premier descripteur de prédicat dans le texte
        CD ← trouve_premier_concept_descripteur(TEXTE)
        Tant Que CD ≠ ∅ et pas_fini(TEXTE) faire
            P ← est_prédicat(CD)
            num_instance[P]++
            Si défini_avec_option(P) alors
                Ob ← recherche_objet(CD,P,TEXTE)
            else
                Ob ← SANS_OBJET
            Finsi
            Si Ob ≠ ∅ alors
                // étiquetage du descripteur par le nom et le
                // numéro d'instance du concept
                étiquetage_descripteur(CD,P,num_instance[P],Ob,TEXTE)
                // recherche et étiquetage des concepts reconnus comme
                // instance d'une option de P
                recherche_et_étiquetage_options(CD,P,num_instance[P],TEXTE)
                num_instance[P]++
            Fin Si
            // recherche du descripteur de prédicat suivant dans le texte
            CD ← trouve_nouveau_concept_descripteur(TEXTE)
        Fin Tant Que
        // test pour vérifier s'il reste encore des règles applicables
        // (aucune règle n'est applicable si en une passe le texte reste inchangé)
        TEXTE_FIN ← TEXTE
        Si TEXTE_FIN = TEXTE_INIT alors
            est_applicable = FAUX
        Finsi
    Fin Tant Que
    // traitement des descripteurs restants
    CD ← trouve_premier_concept_descripteur(TEXTE)
    Tant Que CD ≠ ∅ faire
        étiquetage_prédicat_vide(CD)
        CD ← trouve_nouveau_concept_descripteur(TEXTE)
    Fin Tant Que
Fin étiquetage_prédicatif
```

Figure 7.6 : Algorithme de traitement des règles prédicatives de **SYGET**

Exemple 7.20 (Étiquetage prédicatif)

Le texte de l'exemple 7.17, obtenu à partir de l'extrait de corpus original

> *projet achat 04/2003 bmw serie 3 30ke*

devient, après l'exécution du module d'étiquetage prédicatif, le texte suivant :

```
<C_PROJET instance="1">
   <C_DC_PROJET>projet</C_DC_PROJET>
</C_PROJET>
<C_PROJET instance="1" arg="objet">
   <C_ACHAT instance="1">
       <C_DC_ACHAT>achat</C_DC_ACHAT>
   </C_ACHAT>
</C_PROJET>
<C_PROJET instance="1" arg="date">
    <C_ACHAT instance="1" arg="date">
       <C_DATE>
           <C_DATE_FLOUE_1>04/2003</C_DATE_FLOUE_1>
       </C_DATE>
   </C_ ACHAT>
</C_PROJET>
   <C_ACHAT instance="1" arg="objet">
          <C_VEHICULE><
              C_AUTO>
                  <C_MARQUE_AUTO>bmw</C_MARQUE_AUTO>
                  <C_MODELE_AUTO>serie 3</C_MODELE_AUTO>
              </C_AUTO>
          </C_VEHICULE>
   </C_ACHAT>
<C_PROJET instance="1" arg="montant">
   <C_ACHAT instance="1" arg="montant">
       <C_SOMME>30ke</C_SOMME>
   </C_ACHAT>
</C_PROJET>
```

Le texte est balisé par l'application des règles prédicatives suivantes :

```
C_PROJET :   C_ DC_PROJET
             C_ PROJET_objet : C_ACHAT | C_VENTE | C_IMMOBILIER |
                               C_VEHICULE | C_ACTION_BANCASS
             C_ PROJET_datation :: C_DATE
             C_ PROJET_localisation : C_LIEU
             C_ PROJET_montant :: C_SOMME

C_ACHAT :    C_DC_ACHAT
             C_ACHAT_objet : C_IMMOBILIER | C_VEHICULE | C_PROD_BANCASS
             C_ACHAT_datation : C_DATE
             C_ACHAT_localisation : C_LIEU
             C_ACHAT_montant : C_SOMME
```

7.3.3 Conclusion

À l'issue de l'exécution du module d'étiquetage sur un texte, le résultat produit par **SYGET** est un fichier XML dans lequel les concepts et relations décrits dans l'ontologie d'extraction apparaissent clairement grâce aux balises. Ce fichier peut être consulté directement par une application acceptant le format XML comme par exemple un navigateur Internet (cf. figure 7.7) ou être traité par le module de recueil des informations.

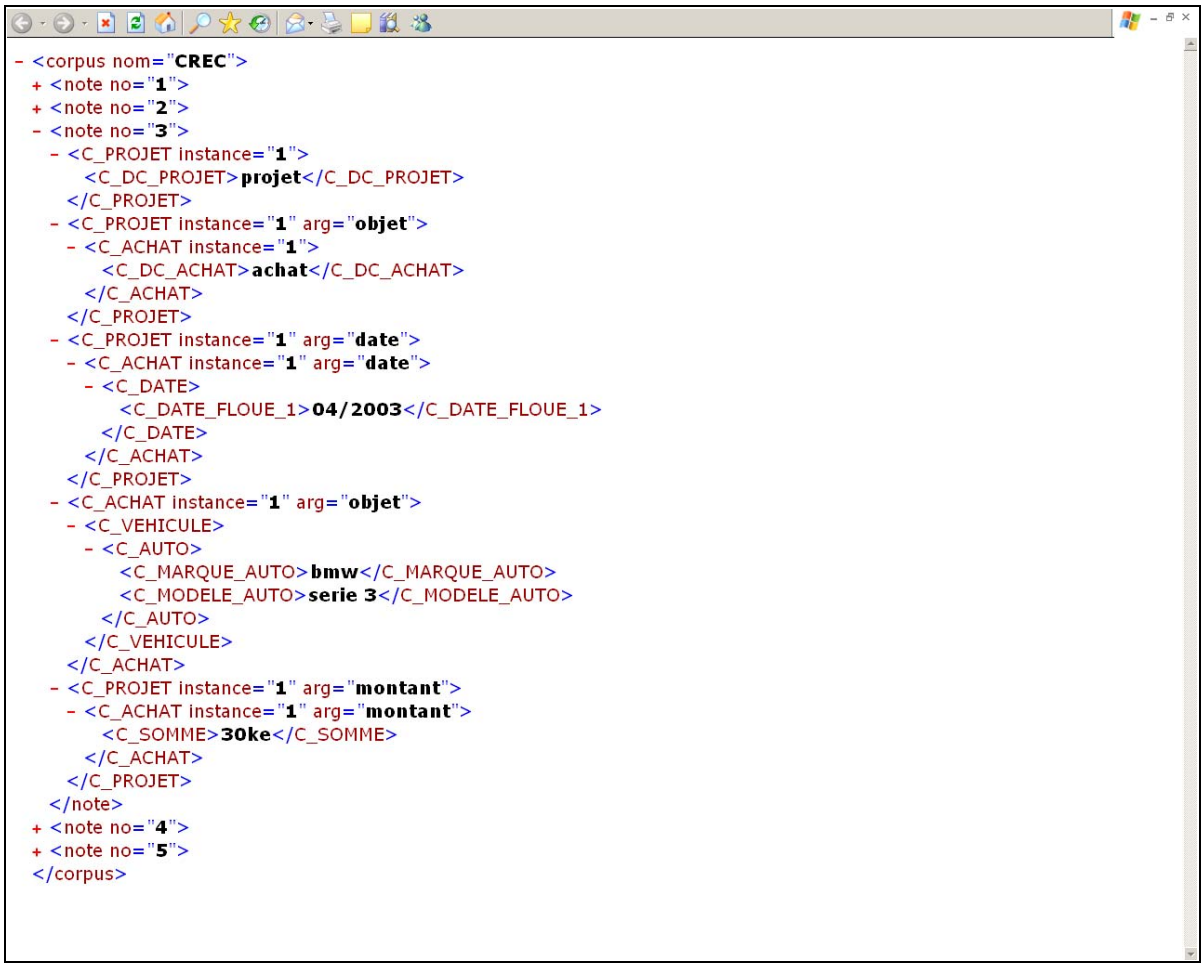

Figure 7.7 : Fichier XML issu de **SYGET** consulté avec Internet Explorer 6

7.4 Module de recueil des informations

L'ontologie d'extraction étant fondée sur des concepts décrivant les informations à rechercher, ces informations sont marquées dans le texte lorsqu'elles sont présentes. Le module de recueil prend en entrée le nom des concepts à rechercher et renvoie en résultat toutes les instances de ces concepts identifiées dans le texte par les balises. Il s'agit en premier lieu des concepts définis par des prédicats (correspondant aux types d'information spécifiés par les experts). Néanmoins, il peut également s'agir de concepts décrits autrement en fonction de ce que l'on désire extraire (par exemple il est possible d'extraire tous les prêts bancaires cités dans le corpus en repérant les instances du concept C_PRET).

Lorsqu'un concept C_A à rechercher est défini par un prédicat P_A, le module de recueil des informations renvoie un ensemble de formulaires (cf. exemple 7.21). Pour chaque instance de C_A, le système produit un formulaire nommé « A » et dont les champs correspondent aux différents arguments de P_A. Un champ supplémentaire indique le numéro de l'instance. Les concepts décrits par une règle constitutive étendue sont traités de la même manière (cf. exemple 7.22).

À l'instar des formulaires complexes des conférences MUC (cf. sections 1.3.2 et 5.1.1), les valeurs des champs sont des entités lexicales issues du corpus ou des pointeurs vers d'autres formulaires. Les champs d'un formulaire sont valués par des pointeurs vers d'autres formulaires :

- Lorsque la valeur de l'objet pour une instance I_{C_A} de C_A est égale à une instance I_{C_B} d'un concept C_B défini par un prédicat ou une règle constitutive étendue, le champ objet du formulaire correspondant à I_{C_A} est valué par une référence au formulaire correspondant à I_{C_B} (nom « B » et numéro d'instance de I_{C_B}) ;
- Lorsque la valeur d'une option (ou d'un attribut) opB_j pour une instance I_{C_B} d'un concept C_B, value une option opA_i pour une instance I_{C_A} de C_A (c'est-à-dire dans les cas où I_{C_B} value l'objet pour I_{C_A} et $opA_i == opB_j$), le champ opA_i du formulaire correspondant à I_{C_A} est valué par un pointeur vers le champ opB_j du formulaire correspondant à I_{C_B} (`[référence de` I_{C_B}`]@ nom de` opB_j).

Le format de sortie est un fichier texte structuré. Ce type de fichier permet facilement de transcrire les résultats dans d'autres formats (base de données ORACLE, formulaires Web, etc.).

Un sous-module optionnel permet de renvoyer les dates sous un format unique et de transformer les dates floues en dates complètes grâce à des calculs réalisés à partir d'une date externe aux séquences textuelles traitées par le module d'étiquetage (date courante, date de rédaction du texte, etc.).

Remarque

Le système peut s'arrêter après l'exécution du module d'étiquetage, sans effectuer le recueil des informations : le format XML des fichiers produits par **SYGET** permet en effet

d'envisager leur exploitation par des outils traitant des documents XML (transformations XSL[2], interrogations par des requêtes XQuery[3], etc.). Il est également possible d'utiliser le système comme un système de réécriture permettant de reformater des documents représentables par un modèle (documents HTML par exemple).

De même les modules d'étiquetage et de recueil peuvent être exécutés sur un texte sans le recours préalable au module de prétraitement s'il s'avère inutile.

Exemple 7.21 (Recueil des informations 1)

À partir du texte de l'exemple 7.20, issu de l'extrait de corpus original

```
projet achat 04/2003 bmw serie 3 30ke
```

le module de recueil des informations produit les formulaires suivants :

```
[ ACHAT
        id instance   :   1
        descripteur   :   achat
        objet         :   bmw serie 3
        date          :   04/2003
        localisation  :   Ø
        montant       :   30ke
]

[ PROJET
        id instance   :   1
        descripteur   :   projet
        objet         :   [ACHAT id_instance=1]
        date          :   [ACHAT id_instance=1]@date
        localisation  :   Ø
        montant       :   [ACHAT id_instance=1]@montant
 ]
```

[2] XSL : `http://www.w3.org/Style/XSL/`

[3] XQuery : `http://www.w3.org/TR/xquery/`

Exemple 7.22 (Recueil des informations 2)

À partir du texte de l'exemple 7.18, issu de l'extrait de corpus original

```
proposition pret immo de 75kf sur 24mois 5,70%
```

le module de recueil des informations produit le formulaire suivant :

```
[ PRET
        id instance   :   1
        descripteur   :   pret immo
        durée         :   24mois
        taux          :   5,70%
        montant       :   75ke
]
```

Chapitre 8

Expérimentations et évaluations

Présentation

Nous avons procédé à plusieurs expérimentations afin de valider la méthode **MeGET** et les idées mises en œuvre dans **SyGET**. Notre principale expérience a été réalisée sur le corpus [CREC], un corpus constitué de Notes de Communication Orale prises lors d'entretiens passés entre des clients et des employés d'une banque. Une deuxième expérience a été effectuée sur un second corpus de Notes de Communication Orale issues de conversations téléphoniques (corpus [Phoning]). Afin d'évaluer la possibilité d'utiliser notre méthode sur d'autres types de textes, nous avons également réalisé une expérimentation sur un corpus issu d'une liste de diffusion par courrier électronique (corpus [LN]).

Nous décrivons d'abord les critères d'évaluation que nous avons définis (section 8.1), puis nous présentons les expérimentations sur les corpus [CREC] (section, 8.2) et [Phoning] (section 8.3). Ensuite, nous analysons les résultats obtenus sur ces corpus de Notes de Communication Orale (section 8.4). Nous terminons ce chapitre par la présentation de l'expérimentation effectuée sur le corpus [LN] (section 8.5).

8.1 Critères d'évaluation

Nos expérimentations sont évaluées en comparant les résultats du processus avec les résultats d'une étude réalisée par des experts en instanciant manuellement les prédicats correspondant aux informations recherchés (méthode d'évaluation proche de celles utilisées lors des campagnes MUC, cf. section 1.3.2).

Le résultat de l'exécution de **SyGET** sur un corpus est un ensemble d'instances des concepts recherchés. Les informations à rechercher étant modélisées par des concepts décrits chacun par un prédicat, les instances de concepts extraites lors de la phase de recueil des informations correspondent à des instances de prédicats.

Un prédicat possédant un ou plusieurs arguments, nous définissons trois degrés de validité en fonction de la manière dont sont valués ces arguments :

1. Une instance d'un prédicat est dite ***valide*** si ses arguments sont valués par les valeurs correctes attendues (cf. figure 8.1) et *non valide* si au moins un argument possède une valeur erronée ;
2. Une instance valide l'est ***totalement*** si tous ses arguments sont valués, et ***partiellement*** si au moins un de ses arguments n'est pas valué ;
3. Une instance valide partiellement est dite ***incomplète*** lorsque au moins un argument n'est pas valué alors que l'information est présente dans le corpus, dans le cas contraire elle est ***complète*** (la totalité des arguments non valués est due à l'absence des informations correspondantes dans le texte).

L'ensemble des instances *valides totalement* et des instances *valides partiellement complètes* sont englobées sous le terme « instances *valides complètement* ». Il s'agit de toutes les instances pour lesquelles tous les arguments qui peuvent l'être (c'est-à-dire tous ceux pour lesquels il existe une valeur dans le texte) sont correctement valués. La réunion des instances *valides complètement* et des instances *valides partiellement incomplètes* regroupe ainsi toutes les instances *valides* extraites par le système (cf. figure 8.1). Cette dichotomie « instances valides complètement / instances valides partiellement incomplètes » permet de mieux évaluer la complétude du système, c'est-à-dire sa capacité à extraire pour une instance, toutes les informations qui sont présentes dans le texte.

Du point de vue de l'Extraction d'Information, le taux prépondérant pour évaluer notre méthode est le taux de précision (cf. section 1.3.2) car l'obtention d'informations erronées se révèle très pénalisante. En effet des informations erronées peuvent nettement fausser le résultat des études, analyses et conclusions qui seront réalisées ultérieurement. Le taux de précision correspond ici au nombre d'instances valides par rapport au nombre d'instances trouvées.

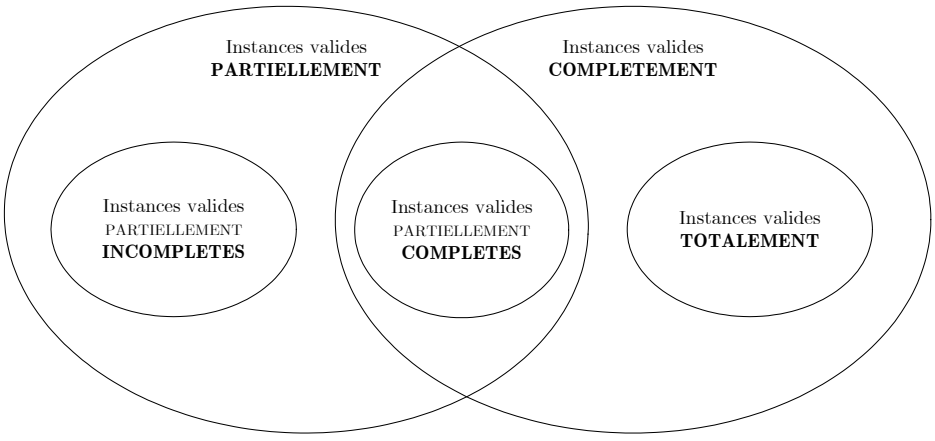

Figure 8.1 : Instances valides

8.2 Corpus [CREC]

8.2.1 Présentation du corpus

8.2.1.1 Présentation générale

Le corpus [CREC] est formé de Notes de Communication Orale appelées Comptes Rendus d'Entretiens Commerciaux (CREC) qui relatent les entretiens passés entre les clients et les employés (banquiers) des diverses agences d'une banque française, le *Crédit Mutuel Loire-Atlantique Centre Ouest* [1].

Il s'agit de notes écrites directement par le banquier lors d'entretiens avec les clients (conversations réalisées au guichet d'une agence la plupart du temps). Lors de ces entretiens, les clients informent le banquier de décisions prises (fermeture de compte, ouverture de livret, etc.), de projets envisagés à court, moyen ou long terme (achat d'une maison, achat d'une voiture pour un membre de la famille, etc.) ou des changements dans leur situation familiale ou professionnelle (naissance d'un enfant, changement d'emploi, etc.). Chaque Compte Rendu d'Entretien Commercial (un CREC) est une Note de Communication Orale qui rapporte le contenu d'un entretien passé entre le rédacteur et un unique client.

Les CRECs sont acheminés mensuellement des différentes agences du CMLACO vers le siège social régional qui centralise ces données et les intègre au corpus. Aussi la taille du corpus croît d'une centaine de milliers de CRECs par an. Fin 2003, le corpus dépassait le million de CRECs alors que nous n'en disposions que d'un peu moins de 800000 au début de l'année 2001.

Les Notes de Communication Orale formant le corpus [CREC] étant librement remplies, elles contiennent des informations qui ne sont pas toujours en accord avec la législation sur la création de fichiers informatiques. Il s'agit d'informations de type religieux, politique ou racial, et des appréciations plus ou moins subjectives des auteurs sur le physique, le caractère, l'intelligence ou encore la santé des individus, etc. Aussi, avant tout traitement, un outil, développé par IBM France, élimine les informations dont le contenu n'est pas en conformité avec les règles de la CNIL (*Commission Nationale de l'Informatique et des Libertés*). Ce processus peut provoquer des coupures abruptes dans les textes. Par exemple, le texte « `DEMANDE PRET IMMO MR ANDRE MAIS PARAIT EN MAUVAISE SANTE` » sera transformé en « `DEMANDE PRET IMMO MR ANDRE MAIS PARAIT` » (l'appréciation sur l'état de santé du client est éliminée).

Le million de Notes de Communication Orale constituant le corpus correspond à un total d'environ 20 millions de mots.

[1] Le *Crédit Mutuel Loire-Atlantique Centre-Ouest* (*Crédit Mutuel LACO* ou *CMLACO*) est une des fédérations régionales du *Crédit Mutuel*, banque mutualiste française.

8.2.1.2 Caractéristiques linguistiques

Chaque CREC se présente sous la forme d'une ligne constituée d'un en-tête numérique et d'un champ texte. La figure 8.2 présente un extrait du corpus [CREC].

L'en-tête est composé du mot « *CREC* » suivi d'une suite de chiffres respectant un format spécifique. Cette suite de chiffres exprime des dates (date de l'entretien et date d'incorporation du CREC au corpus) et des numéros de référence (numéro de client et de CREC).

Le champ texte correspond au texte saisi par les employés et rendant compte de leur entretien avec les clients. La taille du texte varie d'un CREC à un autre : de quelques mots à plus d'une trentaine. Le texte est écrit en français. Il ne suit pas de formalisme particulier (texte libre) et se caractérise par plusieurs particularités linguistiques (cf. section 2.2) :

- Texte entièrement écrit en majuscules (absence d'accents) ;
- Syntaxe fortement dégradée : phrases très simplifiées (souvent réduites à des groupes nominaux), peu de verbes, faible respect de la ponctuation et des règles de segmentation des mots (plusieurs mots se retrouvent collés les uns aux autres). La qualité de la syntaxe dépend du rédacteur. Il n'y a pas de règle de rédaction spécifiques à ces Notes de Communication Orale (pas de rédaction contrôlée) ;
- Présence importante de fautes orthographiques : ces fautes sont extrêmement nombreuses et variées : oubli de lettre, changement d'une lettre par une autre, syllabes inversées, insertion d'un ou plusieurs espaces à l'intérieur des mots ("*SOLD ERI E*" pour "*SOLDERIE*"). Pour un même mot il n'est pas rare de trouver des combinaisons de ces différentes fautes ;
- Présence de très nombreuses abréviations. À l'instar de la syntaxe, les abréviations dépendent du rédacteur et ne sont pas standardisées. Néanmoins de nombreuses abréviations sont communes à plusieurs CRECs ;
- Présence de nombreuses entités nommées sous leur forme complète ou abrégée (acronymes) : il s'agit de noms de personnes (nom et prénom), d'entreprises (concurrents du *Crédit Mutuel* ou constructeurs automobiles par exemple) ou d'institutions ("*Caisse Nationale d'Assurance Maladie* ", "*Banque de France* ", etc.) ainsi que des lieux (adresses complètes ou partielles, nom de commune ou de département, etc.) ;
- Présence d'un grand nombre de termes spécifiques (ainsi que des abréviations de ces termes) au domaine bancaire ou appartenant à un vocabulaire particulier au *Crédit Mutuel* (principalement des noms de produits bancaires). Il peut s'agir de termes existant en dehors du domaine ou des textes mais avec un sens totalement différent de leur signification usuelle (par exemple "*titane* " désigne ici un plan épargne et non un métal) ;
- Beaucoup de dates sont exprimées sous des formes variées d'expressions numériques (JJ/MM/AAAA, JJ/MM/AA, JJMMAA, MM.AAAA, JJ MM, etc.).

```
CREC010000000225.04.1999H13001.01.99991999-12-07-17.04.07.252393   0      EU CE JOUR AU TEL
POUR CHGT DE MANDATAIRE MME RIVALU FABIENNE EN  REMPLACEMENT DE MR RACHUT.
+ DEMANDE DE RENEGO PRET OGEC.
CREC010000000217.06.1999U62401.01.99991999-12-07-17.04.07.258894   0      PRPOSITION PRET
TRAVAUX TRCM ET 0.30                                  SOIT 4,60 SUR 7 QNS 200000 FRS
CAUTION UDOGEC
CREC010000000207.12.1999U13001.01.99991999-12-07-17.04.07.256476   0      REGUL CE JOUR DU
COMPTE A DEBITER POUR LE PRET DE 200KF (PRIS PAR ERR EUR SUR OGEC KERMESSE) LE CTE PAYEUR EST
LE 05811131.                 VU AVEC MR RACHUT CE JOUR
CREC010000001229.06.1999U13001.01.99991999-06-29-16.03.43.720427   0      VU CE JOUR POUR
PROJET INVESTISSEMENT IMMOBILIER EN 20000 POUR 415KF  SUR 10 12 OU 14 ANS.
CREC010000002413.11.1999K10101.01.99991999-11-13-08.59.50.416317   0      MAJ DES
PROCURATIONS
CREC010000002823.10.1998O17101.01.99991998-10-23-16.36.41.032232   0      CHGT DE BUREAU LE 6
10 1998
CREC010000001219.12.1997P10101.01.99992000-05-25-17.13.24.561117   0      MAJ PROCURATIONS
CREC010000004511.12.1998M13001.01.99991998-12-11-14.46.43.544591   0      VENTE CM MONETAIRE
32KF POUR VIT SUR CSL 03997136.
CREC010000007810.06.1998Q13001.01.99991998-12-11-09.03.40.826688   0      VU POUR FAIRE LE
POINT TARIFICATION + EPARGNE. DANS L'ATTENTE DES PRO GRAMME CPAM POUR INVESTIR DANS MAT
INFORMATIQUE.                      PROJET ACHAT VEHICULE , PROPOSITION PRET DE 65KF SUR 24
MOIS A 5.70%.
CREC010000009820.01.1998713001.01.99991998-01-20-15.22.10.423319   0      RBST DE 300KF DE
BONS 2 ANS + 22KF D'INTS. AVIRER SUR LE LIVRET BLEU.
CREC010000002318.12.1999L13001.01.99991999-12-18-13.08.37.852204   0      VU CE JOUR POUR LE
POINT BESOIN CONSEIL PLCMT TRESORERIE VU LE LIVRET BLEU + SICAV MONETAIRE. ETANT DONNE LES
TAUX ACTUELS LA RESTRUCTURATION DES MUTUELLE ET DE LA FISCALITE DES ASSOCIATIONS LAISSER SUR
LIV BLEU.
CREC010000006515.11.1997F10101.01.99991997-11-15-12.32.40.367741   0      MAJ DES MANDATAIRES
CREC010000001010.02.1998K17101.12.19981998-12-12-09.26.10.126624   0      COMME PREVU FAIT
PRET VOITURE DE 30000F
CREC010000003218.02.1998K10101.01.99991998-12-12-09.26.10.128893   0      SUR DEMANDE DU
CLIENT ET EN ACCORD AVEC PIERRE ALAIN. CLOTURE DU PEL A 6 % ET REOUVERTURE D'UN NOUVEAU PEL A
4.25 CAR PROJET
CREC010000000512.12.1998K17101.05.19991998-12-12-09.26.10.130569   0      VU POUR SIMUL PRET
AMENAGT HABITAT
CREC010000001127.05.1998K17101.05.19992000-02-18-12.26.04.656131   0      VUE MME POUR RENS
REMPLAC CREDIMEDIAT PAR PREFERENCE
CREC010000002313.11.1999M13001.01.99992000-02-18-12.26.04.615663   0      VU CE JOUR POUR
ETUDE MODIF PEF EN COURS A VOIR EN DEBUT 2000 SUITE A PRET MAISON ECHU EN JANV 20000.
VU CTE ACTIF ET TARIFICATION
CREC010000005418.02.2000M13001.01.99992000-02-18-12.26.04.6686484  0      PROPO PRET ACHAT
RES SECOND PYRENEE A LUCHON 250KF 180 MOIS MODULVARIABLE A 5.70 %.
CREC010000006312.11.1998S79501.01.99991999-06-29-17.50.21.424017   0      PROPO CB,J'AI
ENVOYE UN CONTRAT DOIT REFLECHIR
CREC010000005615.01.2000M13001.01.99992000-05-11-18.00.57.536005   0      LETTRE DEBITEUR
10026.52 F
CREC010000005402.04.1999M13001.01.99991999-04-02-09.47.46.640228   0      VU CE JOUR POUR
ETUDE PRET VOITURE DE 55KF 60 MOIS A 5.70%            LA REPRISE DE 30KF PERMETTRAUN RBST
PARTIEL DU PREFERENCE
CREC010000000314.09.1999N13901.01.99991999-09-14-15.01.42.060608   0      LES ASSEDIC SERONT
VERSEES VERS LE 28091999 3300 FRANCS.              DOIVENT PRENDRE RDV EN OCTOBRE, SINON
REJETS.
CREC010000001207.02.1998M13007.03.19981998-11-12-10.54.57.536726   0      VU CE JOUR POUR
ETUDE RACHAT PRET CNE DE 220KF PROPOSE 5.95 SUR 106 MOMR TRAVAILLE A LA POSTE DE CLISSON ET
MME A L'EDF.                    ACTUEL. AUCUN SALAIRE VERSE A L'AGENCE.
REPONSE DU 01041998 NE DONNENT PAS SUITE A NOTRE OFFRE.
CREC010000005630.04.1999Q05815.09.19992000-06-10-11.11.49.480873   0      RENCONTRE CE JOUR
POUR FAIRE LE POINT                                  OUVERT PEA AVEC ALIMENTATION 30000
LEUR AI CONSEILLE DE NOUS RECONTACTER DES QU'UNE DECISION EST PRISE SUR LEUR AVENIR
PROFESSIONNEL
```

Figure 8.2 : Extrait du corpus [CREC]

8.2.2 Objectifs de l'analyse du corpus

Du point de vue du banquier, les entretiens relatés dans les Notes de Communication Orale du corpus [CREC] représentent une source d'informations sur le client. Le but de l'analyse du corpus est d'en extraire celles qui pourront être utilisées à des fins commerciales. Il s'agit d'informations concernant les achats ou les ventes réalisées par les clients, les changements dans leur situation personnelle (naissance d'un enfant, déménagement) ou professionnelle (nouvel emploi, mutation) et surtout leurs **projets** (immobilier, financier). Les informations extraites viendront alimenter une base de données destinée aux services commerciaux de la banque.

Grâce à ces renseignements, les commerciaux de la banque peuvent connaître les désirs ou besoins éventuels des clients et effectuer des actions commerciales ciblées vers les personnes potentiellement intéressées (par exemple, proposer un prêt auto à quelqu'un qui a exprimé le souhait de changer de voiture). L'intérêt d'actions ciblées est d'abord de réduire les coûts (l'envoi de publipostages à tous les clients représente un coût élevé en regard des résultats obtenus en retour) et également d'éviter de « noyer » le client sous une masse de propositions inintéressantes pour lui. Les résultats de telles actions commerciales sont une source de profit importante pour la banque d'où le grand intérêt porté au traitement des masses de Notes de Communication Orale contenues dans le corpus [CREC].

8.2.3 Détail de l'expérimentation

Conjointement avec des experts du *CMLACO*, nous avons d'abord élaboré une ontologie d'extraction fondée sur les objectifs décrits dans la section précédente. Nous avons ensuite procédé à l'exécution du système **SYGET** sur un extrait du corpus [CREC] en utilisant cette ontologie comme base de connaissances.

Avant tout traitement, un processus de mise en forme propre à ce corpus sépare l'en-tête du champ texte dans chaque CREC. La construction de l'ontologie d'extraction et l'exécution de **SYGET** sont effectuées à partir du contenu des champs textes.

8.2.3.1 Construction de l'ontologie d'extraction

Élaboration de l'ontologie des besoins

Les informations à extraire ont été formalisées avec les experts du *CMLACO* en huit prédicats (projet, achat, vente, etc.). Chacun de ces prédicats définit un concept de l'ontologie des besoins. À partir de ce premier ensemble de concepts, une cinquantaine de concepts supplémentaires ont été définis lors de la phase d'extension du modèle prédicatif. À ces concepts viennent s'ajouter une quinzaine de concepts génériques (dates, valeurs, etc.). À la fin du processus, l'ontologie des besoins se compose de 90 concepts.

Construction de l'ontologie des termes

L'étude termino-ontologique a abouti à la construction de l'ontologie des termes :

- *Prétraitements* : une étude d'un échantillon du corpus [CREC] de 10000 Notes de Communication Orale (soit 1% du corpus et 200000 mots) a abouti à l'écriture des règles de réécriture (cf. section 7.1.1). Les textes composant l'échantillon ont été choisis de manière équilibrée dans l'ensemble du corpus : les 3000 premiers ont été pris dans le premier tiers du corpus, les 4000 suivants aléatoirement autour du milieu du corpus et les 3000 derniers dans le troisième tiers. Les règles ont ensuite été fournies au module de prétraitement du système. L'étude de l'échantillon et la définition des règles de prétraitements ont été réalisées en deux semaines. L'exécution des prétraitements eux-mêmes n'a duré que quelques heures.

- *Extraction et sélection des termes* : le logiciel ANA a été appliqué sur le sous-corpus prétraité issu de l'échantillon précédent. Cette application a permis d'extraire 15000 candidats-termes. Après filtrage (élimination des parasites, regroupement des variantes morphologiques de mêmes termes, etc.), l'étape de sélection menée conjointement avec des experts de la banque a permis de retenir 1300 termes. Nous avons ensuite enrichi manuellement ce premier ensemble de termes grâce à des ressources terminologiques propres au *CMLACO* se présentant sous la forme de documentations techniques indexant l'ensemble des produits bancaires de cette banque. Sur les 350 termes contenus dans ces documents techniques, nous en avons retenu 200 qui sont venus s'ajouter aux 1300 termes précédents. La terminologie du modèle est ainsi constituée de 1500 termes.

- *Conceptualisation des termes et définition de réseaux de concepts* : ces deux phases ont abouti à la description de 63 nouveaux concepts (42 concepts de base et 21 concepts définis dans des réseaux). Elles ont également conduit à la lexicalisation de 30 concepts de l'ontologie des besoins (chacun de ces 30 concepts est déterminé comme conceptualisant un ou plusieurs termes du corpus). Finalement l'ontologie des termes comprend donc 93 concepts.

L'ontologie des termes a ensuite été unifiée à l'ontologie des besoins. Le résultat de ce processus est une ontologie d'extraction composée de 130 concepts et décrite par autant de règles formelles. Le temps total passé à l'élaboration de cette ontologie a été d'environ sept semaines.

8.2.3.2 Exécution du système SYGET

Les règles constitutives et prédicatives décrivant l'ontologie d'extraction sont fournies au système **SYGET**. Avant de procéder à une réelle expérimentation du processus d'extraction, nous avons procédé avec les experts à des expériences sur de courts extraits du corpus (extraits de quelques dizaines de lignes) afin de calibrer le système. Ces expériences ont permis de paramétrer au mieux, pour ce corpus, les règles prédicatives, conjonctives et disjonctives de la base de règles en jouant sur la taille, le sens et le contenu des fenêtres de recherche. Moins de deux semaines ont été nécessaires pour réaliser ces réglages.

À ce point, le système est prêt à analyser le corpus. Nous avons donc procédé à l'expérimentation du système à proprement parler en l'exécutant sur un échantillon de 10000 Notes de Communication Orale prises au hasard dans le corpus. Afin d'évaluer la qualité réelle des résultats produits par le système, nous avons choisi un échantillon différent de celui utilisé pour l'étude termino-ontologique.

8.2.4 Résultats

Nous avons évalué les performances du système en nous intéressant aux informations désignées comme étant les plus importantes : les **projets**. La notion de projet est conceptualisée dans l'ontologie par le concept C_PROJET, défini par le prédicat P_PROJET. Un projet identifié dans le texte correspond donc à une instance du concept C_PROJET et par conséquent à une instance du prédicat P_PROJET.

Dans l'échantillon étudié, les experts ont noté la présence de 659 projets. À partir de ce même échantillon, le système **SYGET** a extrait 636 instances de P_PROJET (96,5 % des projets). Les instances de prédicat vides (aucun argument valué en dehors du descripteur) ne sont pas prises en compte ni par les experts ni par le système.

Parmi les 636 instances du prédicat P_PROJET détectées, 604 sont valides, soit 95 % des instances trouvées (taux de précision). Seulement 5 % des instances extraites sont non valides (32 instances).

Le taux de rappel (nombre d'instances valides trouvées par rapport à toutes celles présentes) obtenu par le système est égal à 91,6 % et s'avère être très satisfaisant, surtout en regard de la qualité lexicale et syntaxique des textes analysés. Il signifie que peu d'instances du prédicat recherché sont *oubliées* par le système. Nous étudions maintenant les instances repérées du point de vue des critères exposés à la section 8.1. Les résultats de cette étude sont résumés par la figure 8.3.

526 instances valides ne le sont que partiellement : pour 87 % des instances valides, toutes les informations recherchées ne sont pas trouvées. Ce phénomène est beaucoup plus lié à l'incomplétude du corpus (les informations nécessaires pour valuer tous les arguments de certaines instances de prédicats sont absentes du corpus) qu'à des manquements du système. En effet 79,6 % des instances valides partiellement sont valuées complètement par le système, signifiant que toutes les informations présentes dans le corpus pour chacune de ces instances ont été extraites. Ainsi 82,2 % des instances valides le sont complètement.

L'incomplétude du corpus s'explique par le fait que les rédacteurs, lors d'un entretien synthétisé par un CREC, n'obtiennent pas toujours la totalité des informations concernant un sujet donné (par exemple lors d'une discussion traitant d'un futur achat, le client ne communique pas la date de ce futur achat). Mais, bien que ces absences d'informations soient responsables de la grande majorité des cas d'instances valides partiellement, les manquements dus au système (instances valides partiellement incomplètes) ne sont pas négligeables car ils sont à l'origine d'un peu plus de 20 % des cas d'instances valides partiellement (cf. section 8.4). Ce chiffre permet de calculer que 17,8 % des instances valides extraites par le système sont partiellement incomplètes (cf. figure 8.7, page 186).

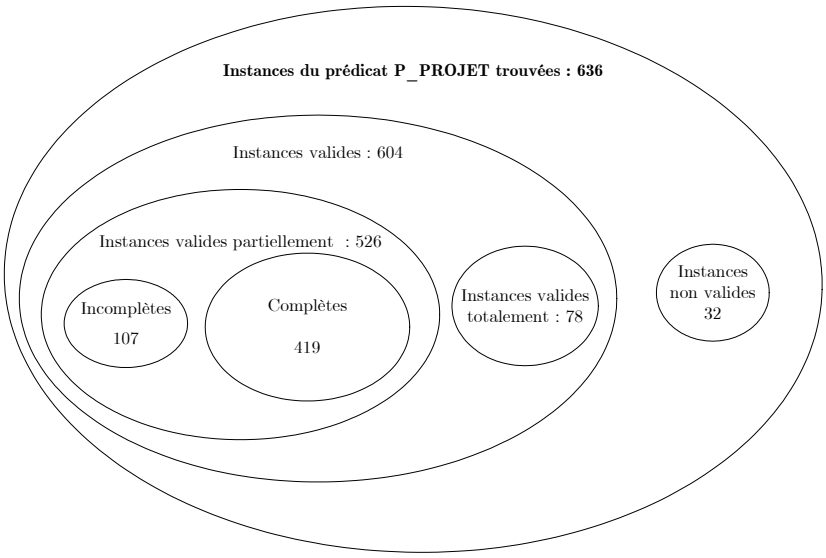

Figure 8.3 : Résultats de l'expérimentation sur le corpus [CREC]

8.3 Corpus [Phoning]

8.3.1 Présentation du corpus

Le corpus [Phoning] est composé de Notes de Communication Orale réalisées lors de conversations téléphoniques. Ces conversations correspondent à des enquêtes réalisées par *phoning*. Le *phoning* est une activité de marketing usuellement définie comme *une prise de contact téléphonique sur un coeur de cible marketing.*

Chaque conversation est réalisée entre un opérateur d'une plate-forme de téléphonie d'une banque, et un interlocuteur qui peut être un client de la banque ou une personne considérée comme potentiellement intéressée par des produits de la banque.

Il s'agit pour les opérateurs de proposer à chacun de leurs interlocuteurs des produits bancaires (crédits, assurances, etc.) et d'enregistrer leur intérêt (dans l'immédiat ou dans le futur) pour ces propositions ou au contraire leur refus.

Chaque texte du corpus [Phoning] correspond à une conversation et est composé d'une partie numérique identifiant la conversation (numéro de référence et date de la conversation) et d'une partie texte constituée de notes relatant son contenu. Ces notes sont saisies directement au clavier lors de la conversation grâce à un outil informatique dédié à l'activité de phoning[2].

Le corpus [Phoning] est constitué de 8000 Notes de Communication Orale correspondant à un total d'environ 165000 mots.

[2] De nombreux outils de ce genre sont présentés sur le site de la société Phonetic : `http://www.phonetic.fr`

8.3.2 Expérience

Le but de l'expérience sur le corpus [Phoning] est d'en extraire les refus des clients. Pour chaque refus d'une proposition, il s'agit de trouver le type du produit proposé et la cause de ce refus (pas de besoin immédiat, produit meilleur chez un concurrent, prix trop élevé, hostilité à la consommation de ce type de produit, etc.).

L'objectif est de collecter des informations sur la perception du produit par les clients (pertinence et qualité des prestations, prix, adéquation avec certaines catégories socioprofessionnelles) et sur les produits similaires chez la concurrence (lorsque le refus est motivé par l'existence d'un produit meilleur ailleurs) afin d'améliorer les offres commerciales de la banque.

Nous avons exécuté le système **SYGET** sur le corpus [Phoning] pour qu'il identifie les instances d'un prédicat P_REFUS défini dans l'ontologie d'extraction construite avec les experts. Ce prédicat a pour objet des concepts exprimant un produit bancaire ou une proposition d'un produit, et possède une option exprimant la cause du refus. L'ontologie d'extraction élaborée à partir de ce prédicat est composée d'une soixantaine de concepts et d'environ 700 termes (dont un grand nombre provient des lexiques de produit bancaires).

Les temps de construction de l'ontologie d'extraction et de calibrage du système pour le corpus [Phoning] sont, théoriquement, sensiblement les mêmes que ceux de l'expérience sur le corpus [CREC]. Néanmoins, le recours à des concepts déjà définis pour le corpus [CREC] (concepts relatifs au domaine bancaire et particulièrement à la notion de produit bancaire et concepts génériques) ainsi que l'expérience acquise de l'expérimentation précédente (cf. section 8.2), ont permis de diminuer la durée de la phase d'écriture des règles décrivant les concepts (gain de deux semaines de travail environ).

8.3.3 Résultats

Nous avons étudié un échantillon du corpus [Phoning] contenant 600 refus de proposition (repérés manuellement par les experts) afin que les résultats soit comparables à ceux du corpus [CREC] (même ordre de grandeur au niveau du nombre d'instances de prédicat à détecter). La figure 8.4 présente les résultats obtenus.

Sur cet échantillon, 568 instances du prédicat P_REFUS sont détectées par le système (soit 94,6 % des refus). Parmi ces instances, 531 sont valides. Les taux de rappel et de précision du système sur ce corpus sont donc respectivement de 88,3 % et 93,5 %.

48 % des instances valides le sont totalement, c'est-à-dire que la cause du refus a été correctement trouvée. Parmi les 276 instances valides partiellement (52 %), 72 % sont complètes, c'est-à-dire que la cause du refus n'est pas exprimée dans les textes. Les 28 % restant (correspondant à 13,4 % de l'ensemble des instances valides) sont partiellement incomplètes : la cause du refus est indiquée dans le texte mais pas détectée par le système (cf. section 8.4). Á partir de ces chiffres, nous calculons que 86,6 % des instances valides le sont complètement (cf. figure 8.7, page 186).

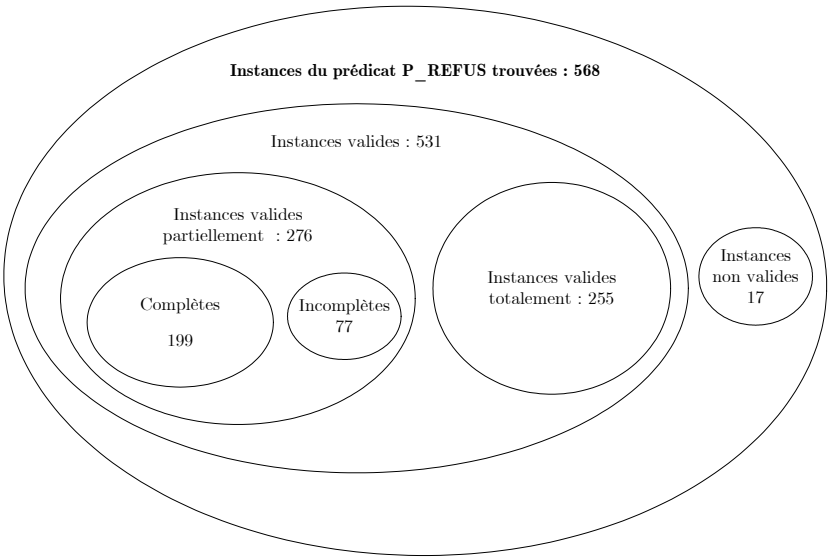

Figure 8.4 : Résultats de l'expérimentation sur le corpus [Phoning]

8.4 Analyse des résultats

L'étude des résultats de nos expérimentations sur des corpus de Notes de Communication Orale (résumés dans les figures 8.6, 8.7 et 8.8, page 186) fait apparaître trois problèmes dont nous analysons ici les raisons : les cas d'instances de prédicats manquantes c'est-à-dire des instances présentes dans le corpus mais non détectées par le système (silence), les cas d'instances non valides (bruit) et les cas de résultats incomplets liés à des manquements du système (instances valides partiellement incomplètes).

8.4.1 Instances manquantes

Une instance de prédicat n'est pas détectée lorsque le terme utilisé par le rédacteur pour exprimer la notion décrite par le prédicat (terme correspondant à la lexicalisation du descripteur du prédicat) n'est pas pris en compte par le système. Ce phénomène survient lorsque ce terme n'a pas été identifié lors de l'analyse terminologique : il s'agit de termes propres à quelques rédacteurs et qui apparaissent très rarement dans le corpus (par exemple des abréviations spécifiques à une seule personne). De tels termes sont si spécifiques et si rarement utilisés qu'ils ne sont pas retenus lors de l'analyse terminologique. Ce phénomène est cependant très limité.

Des instances de prédicat ne sont également pas identifiées lorsqu'une unique instance d'un descripteur décrit plusieurs instances d'un même prédicat. C'est le cas pour un prédicat P_A défini avec un objet, lorsque dans le texte se trouve une séquence composée d'une instance du descripteur de P_A suivi de deux instances de concepts type d'objet de P_A et séparés par une des conjonctions de coordination « *et* » « *ou* ».

Par exemple à partir de la séquence de texte « `<DC_PROJET>`*projet*`</DC_PROJET>` `<C_IMMOBILIER>...</C_IMMOBILIER>` *et* `<C_VEHICULE>...</C_VEHICULE>` », le système

devrait identifier un *projet immobilier* et un *projet de véhicule* mais n'identifie que le *projet immobilier*.

Il s'agit d'un choix volontaire qui évite d'identifier des instances non valides lorsque la conjonction n'est pas utilisée par le rédacteur pour définir deux instances d'un concept C_A mais pour séparer une instance de C_A d'une instance d'un autre concept (comme dans le texte « <DC_PROJET>*projet*</DC_PROJET> <C_IMMOBILIER>...</C_IMMOBILIER> *et* <DC_ACHAT>*acquisition*</DC_ACHAT> ...» qui ne pas désigne pas deux instances de C_PROJET mais une instance de C_PROJET et une instance de C_ACHAT). Ce choix est motivé par la volonté de fournir aux utilisateurs le moins possible d'informations erronées, quitte à en « oublier » certaines.

8.4.2 Instances non valides

Le renvoi d'une instance de prédicat non valide est dû à des positions particulières des informations valuant les arguments de cette instance.

Les trois-quarts des cas d'instances non valides apparaissent lorsqu'une information (exprimée par une instance I_{C_Y} d'un concept C_Y) correspondant pour le rédacteur à la valeur de l'argument *arg1* d'une instance de prédicat I_{P_A} est placée dans le texte après (ou avant en fonction du sens de recherche des valeurs d'options) le descripteur d'une instance I_{P_B} d'un prédicat P_B (ou d'un concept décrit par une règle prédicative étendue) dans lequel il existe un argument *arg2* tel que C_Y $\in$ T_arg2(P_B). Dans cette situation, l'information valuera l'argument *arg2* de I_{P_B} au lieu de valuer l'argument *arg1* de I_{P_A} comme il le faudrait. Ainsi le système renvoie une instance I_{P_A} incomplète et une instance I_{P_B} non valide (cf. exemple 8.1). Ce phénomène existe dans le corpus mais est toutefois peu fréquent (moins d'une vingtaine sur 659 instances de prédicats dans le corpus [CREC], soit moins de 3 %). Les rédacteurs cherchent en effet à limiter l'apparition de ces phénomènes dans les textes car ils les rendent ambigus et difficiles à comprendre même pour un humain.

Exemple 8.1 (Instances non valide et incomplète)

Le texte

> *projet appartement demande pret pour montant total de 200ke*

est transformé par le processus d'étiquetage constitutif en

```
<DC_PROJET>projet</DC_PROJET>
<C_IMMOBILIER><C_APPARTEMENT>appartement</C_APPARTEMENT></C_IMMOBILIER>
demande <C_DF_PRET><C_DC_PRET>pret</C_DC_PRET></C_DF_PRET>
pour montant total de
<C_SOMME>200ke</C_SOMME>
```

À partir du texte étiqueté précédent, le système détectera une instance de C_PROJET et une instance de C_PRET et valuera l'attribut montant de l'instance de C_PRET par "200ke " alors que cette somme est en fait la valeur du montant de l'instance de C_PROJET (montant total du projet immobilier). En conséquence le système renverra une instance de C_PRET non valide et une instance de C_PROJET incomplète.

Les instances non valides restantes sont dues au phénomène suivant : lorsqu'une information n'est pas présente dans le texte pour un argument *arg* d'une instance I_{P_X} d'un prédicat P_X, cet argument ne doit pas être valué. Or dans le cas où une instance d'un concept C_Y ∈ T_arg(P_X) se situe dans le texte à proximité du descripteur de I_{P_X}, elle peut être faussement interprétée par le système comme la valeur de *arg* (cf. exemple 8.2).

Le réglage des paramètres (direction et taille) des fenêtres de recherche des règles permet de limiter sensiblement ces problèmes mais il subsiste quelques cas mal traités par le système.

Exemple 8.2 (Instance non valide)

Le texte

> *projet maison avec 50ke recu par heritage*

est transformé par le processus d'étiquetage constitutif en

> <DC_PROJET>*projet*</DC_PROJET>
> <C_IMMOBILIER><C_MAISON>*maison*</C_MAISON></C_IMMOBILIER>
> *avec* <C_SOMME>*50ke*</C_SOMME> *recu par heritage*

À partir de ce texte étiqueté, le système détectera une instance de C_PROJET et valuera l'option montant de cette instance par "50ke" alors que cette somme n'est par la valeur du montant du projet mais celle d'un héritage.

8.4.3 Incomplétude des résultats

L'analyse des instances valides partiellement incomplètes fait apparaître deux principaux problèmes à l'origine des incomplétudes. Ces deux problèmes sont liés au positionnement des informations dans le texte.

Le premier problème est une conséquence du phénomène provoquant la plupart des cas d'instances non valides, c'est-à-dire lorsque l'information à trouver pour valuer un argument *arg* d'une instance I_{P_X} d'un prédicat est située après (ou avant en fonction du sens de recherche) le descripteur d'une autre instance de prédicat (ou d'un concept décrit par une règle prédicative étendue). Dans ce cas de figure, l'information ne sera pas considérée par le système comme liée à I_{P_X} et l'argument *arg* ne sera pas valué (comme dans l'exemple 8.1).

Le second problème est une question de distance des informations avec les descripteurs de prédicat : le système n'arrive pas à trouver les informations valuant certains arguments d'une instance de prédicat lorsqu'elles se situent loin (en terme de nombre de mots) du descripteur du prédicat (avant ou après), c'est-à-dire en dehors de la fenêtre de recherche déterminée pour ces arguments.

Changer les paramètres pour agrandir les fenêtres de recherche permet au système de prendre en compte ces cas. Mais après quelques expériences dans ce sens, il s'avère que cette modification entraîne une chute de la précision des résultats. En effet lorsque la fenêtre de recherche est très étendue, le processus arrive à valuer plus d'arguments mais souvent de

façon erronée : plus un concept est éloigné d'un descripteur moins sa probabilité d'être liée à ce descripteur est grande. Cette hypothèse se vérifie par une étude de surface de plusieurs corpus et s'explique par la volonté d'un rédacteur de rendre compréhensible son propos en ne disséminant pas dans le texte des informations liées entre elles.

8.4.4 Bilan

Malgré les problèmes d'incomplétude évoqués dans la section 8.4.3, les résultats obtenus sur les corpus [CREC] et [Phoning] ont été jugés très satisfaisants par les utilisateurs désirant exploiter les textes analysés (des banquiers et des commerciaux), surtout en comparaison des résultats obtenus avec les méthodes utilisées auparavant pour extraire de la connaissance de ces corpus (pour le corpus [CREC] il s'agissait d'une méthode fondée sur des règles d'extraction écrites après l'analyse des contextes locaux de termes clés identifiés grâce à un concordancier).

Le temps d'adaptation du processus à un corpus et aux objectifs (temps de création de l'ontologie d'extraction et de réglage des paramètres de recherche) est convenable du point de vue industriel pour une telle application de Fouille de Textes.

En conclusion, les résultats obtenus sont satisfaisants car ils répondent au but que nous avions fixé au début de cette thèse : pouvoir fournir en un temps raisonnable, à partir de Notes de Communication Orale, des informations exploitables pour des utilisateurs des résultats du processus d'extraction.

8.5 Corpus [LN]

8.5.1 Présentation du corpus

Ce corpus est constitué de courriers électroniques issus du bulletin électronique ln@cines.fr[3]. Ce bulletin a pour objectif de favoriser les échanges au sein de la communauté du Traitement Automatique des Langues. Les courriers électroniques ont pour sujet :

- Des appels à communication, annonces et comptes rendus de conférences, de séminaires, d'écoles d'été ;
- Des annonces de parution de livres et de logiciels ;
- Des questions (et réponses) spécifiques concernant logiciels, corpus, etc. ;
- Des offres d'emplois ;
- Des descriptions d'activités et de projets.

Il s'agit de textes rédigés correctement en français ou en anglais (loin des particularités linguistiques des Notes de Communication Orale). Ils se caractérisent par la présence de nombreuses dates (exprimées dans des formats variés), de noms d'institutions, de lieux et de

[3] `http://www.biomath.jussieu.fr/LN/LN-F/`

personnes, de noms d'événements (souvent exprimés par des schémas mêlant mots, nombres et caractères non-alphanumériques) et d'adresses (postales, de courrier électronique, de page Internet). Chaque courriel est constitué d'un champ « `objet` » précisant le sujet du message (appel, offre d'emploi, etc.) et d'un champ texte.

Le corpus [LN] est formé de 158 courriels émis entre juin 2004 et janvier 2005. Sa taille est d'environ 85000 mots.

8.5.2 Expérience

Le but de l'étude du corpus [LN] est d'extraire les appels à communication pour des conférences. Il s'agit de trouver les informations suivantes : le nom de la conférence, le lieu où elle se déroule et ses dates : date de la conférence, date limite de soumission et date de notification d'acceptation aux auteurs.

Techniquement, l'analyse du corpus [LN] consiste à chercher les instances d'un prédicat P_APPEL_SOUMISSION. L'objet de ce prédicat est le nom de la conférence et les autres informations à rechercher (dates, lieu) sont exprimées par des options. Le concept défini par ce prédicat est décrit par la règle suivante :

<C_APPEL_SOUMISSION> ::= {
descripteur = <C_DC_APPEL> ;
objet = <C_NOM_CONFERENCE> ;
localisation = <C_LIEU> ;
date_conference = <C_DATE_CONF> ;
date_soumission = <C_DATE_SOUM> ;
date_notification = <C_DATE_NOTIF>.
}

L'ontologie d'extraction est construite autour de ce concept. Elle est composée d'une cinquantaine de concepts et d'environ un millier de termes (dont un grand nombre de noms de villes, de pays et d'institutions – universités, instituts, laboratoire, etc.). La construction de cette ontologie s'est déroulée en quatre semaines. Cette durée comprend l'écriture des règles de réécriture concernant les concepts génériques ainsi que l'élaboration et l'exécution des prétraitements. Trois jours ont suffi pour la phase de prétraitements, ce temps limité s'expliquant par la qualité du corpus [LN] dans lequel ne se posent pas les problèmes typographiques et morphologiques des Notes de Communication Orale. Toutefois, cette phase de prétraitement n'aurait pu être omise car de nombreuses dates nécessitent une normalisation.

Pour réaliser cette expérience, nous avons dû ensuite régler le système pour le corpus [LN]. D'abord, nous avons modifié le système pour que dans un même courriel, toutes les instances du prédicat ayant le même objet (même nom de conférence) soient considérées comme des occurrences de la même instance. En effet, dans un courriel traitant d'un appel à communication pour une conférence X, ce même appel est souvent répété plusieurs fois dans le texte : chaque évocation de cet appel ne doit pas être interprétée comme une instance du prédicat P_APPEL_SOUMISSION. La possibilité de considérer une instance comme une

entité à part entière ou comme une répétition d'une autre instance, devient une option du système. Enfin, avant d'exécuter le système, nous avons calibré la taille, le sens et le contenu des fenêtres de recherche pour les règles prédicative, conjonctives et disjonctives grâce à des tests sur des extraits du corpus.

8.5.3 Résultats

Le champ objet de chaque courriel traitant d'un appel à communication contient le mot "*appel*", aussi il est facile de discriminer les messages contenant des appels à communication. Néanmoins, les appels pour lesquels le nom de la conférence n'a pas été trouvé (objet du prédicat vide) ne sont pas considérés comme détectés. La figure 8.5 ci-dessous présente les résultats obtenus lors de cette expérimentation.

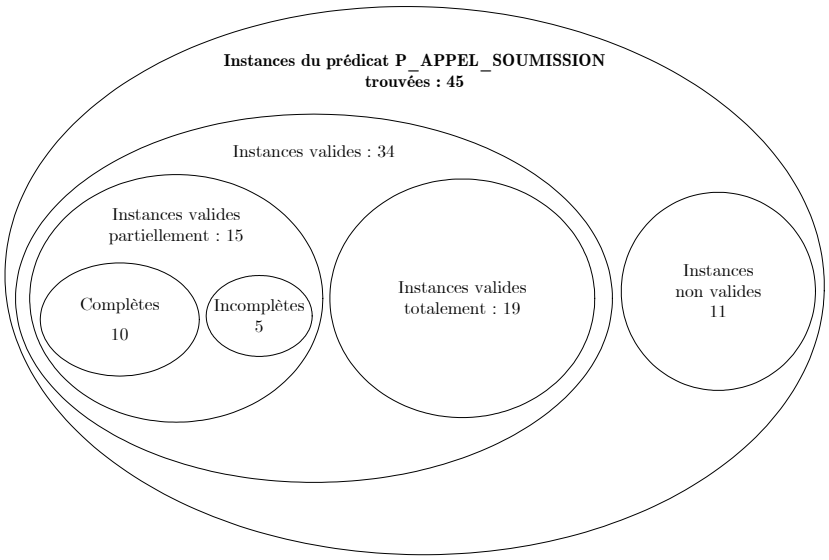

Figure 8.5 : Résultats de l'expérimentation sur le corpus [LN]

Dans le corpus [LN], 53 appels à communication sont présents. Le système en détecte 45 soit 85 %. Parmi les appels détectés, 34 sont valides : en conséquence les taux de rappel et de précision sont respectivement égaux à 64 % et 75,6 %. A l'inverse 24,4 % des instances détectées possèdent un argument instancié avec une valeur erronée (instances invalides) : dans la moitié des cas, il s'agit de l'objet (le système ne trouve pas le bon nom de la conférence à laquelle l'appel fait référence).

56 % des instances valides le sont totalement. Parmi les instances valides partiellement, 67 % sont complètes (l'absence de la date de notification aux auteurs dans les textes a empêché de valuer toutes les options de l'instance de prédicat) et 33 % sont incomplètes.

Á partir de ces chiffres, nous déduisons que 85,5 % des instances valides le sont complètement, les 14,5 % restant étant partiellement incomplètes (cf. figure 8.7, page 186).

8.5.4 Analyse des résultats

Nous constatons qu'en termes de rappel et de précision, les résultats sont ici nettement inférieurs à ceux obtenus sur des corpus de Notes de Communication Orale (cf. figure 8.6). Pour les textes du corpus [LN], ces résultats peuvent s'expliquer par :

- Des caractéristiques linguistiques qui diffèrent de celles des Notes de communication Orale pour lesquelles notre méthode a été développée ;
- La taille plus élevée des textes et leur caractère beaucoup plus syntaxique (et *a contrario* moins sémantique) que les Notes de Communication Orale qui entraine une dissémination des informations tout au long du texte ; cette dissémination posant des problèmes à notre méthode (cf. section 8.4.3) ;
- La taille plus réduite du corpus de mise au point.

Ces résultats apparaissent néanmoins intéressants, notamment au niveau de la complétude des instances trouvées (cf. figures 8.6 et 8.7). En effet, lorsqu'une instance est détectée et valide, dans plus de 50 % des cas, toutes ses options sont valuées par les valeurs correctes : l'ensemble des informations recherchées a été trouvé. Le taux d'instances valides dont les arguments ne sont pas valués en raison de manquements dus au système (taux d'instances valides partiellement incomplètes) est proche de ceux constatés pour les corpus de Notes de Communication Orale. Ces manquements s'expliquent principalement pour le corpus [LN] par le fait que certaines informations peuvent être disséminées dans les textes par les auteurs pour une même instance de prédicat (la date de soumission au début du courriel, celle de notification à la fin, etc.).

Devant les résultats obtenus pour un corpus aux caractéristiques très différentes de celles pour lesquelles notre méthode a été développée, nous sommes encouragés à explorer les solutions mises en œuvre dans **MEGET** et **SYGET** pour le traitement de textes autres que les Notes de Communication Orale. La conclusion de cette thèse présente quelques perspectives dans ce sens.

Corpus	**Instances détectées**	**Taux de Rappel**	**Taux de Précision**	**Instance détectées valides**			
				Totalement	*Partiellement*		
					total	complètes	incomplètes
[CREC]	96,5 %	91,6 %	95 %	13 %	87 %	79,6 %	20,4 %
[Phoning]	94,6 %	88,3 %	93,5 %	52 %	48 %	72 %	28 %
[LN]	85 %	64 %	75,6 %	56 %	44 %	67 %	33 %

Figure 8.6 : Synthèse des résultats (pourcentages – 1)

Corpus	**Taux de Rappel**	**Taux de Précision**	**Instance détectées valides**	
			Complètement	*Partiellement incomplètes*
[CREC]	91,6 %	95 %	82,2 %	17,8 %
[Phoning]	88,3 %	93,5 %	86,6 %	13,4 %
[LN]	64 %	75,6 %	85,5 %	14,5 %

Figure 8.7 : Synthèse des résultats (pourcentages – 2)

Corpus	Instances **trouvées**	Instance **non valides**	Instances **valides**	Instances **valides totalement**	Instances **valides partiellement**		
					total	complète	incomplète
[CREC]	636	32	604	78	526	419	107
[Phoning]	568	17	531	255	276	199	77
[LN]	45	11	34	19	15	10	5

Figure 8.8 : Synthèse des résultats (nombres d'instances)

Conclusion

Conclusion et perspectives

Bilan

Les travaux décrits dans ce manuscrit portent sur l'extraction d'information à partir de textes issus de prises de notes réalisées lors de communications orales (entretiens, exposés, cours, réunions, etc.). Nous regroupons l'ensemble de ces textes sous la dénomination de *Notes de Communication Orale*.

Extraire des informations précises à partir de textes constitue aujourd'hui un enjeu considérable dans de nombreux domaines. C'est particulièrement le cas dans les mondes institutionnel, de l'économie et de l'industrie, dans lesquels les notions de veille technologique et/ou économique sont devenues essentielles, notamment pour l'aide à la prise de décision (définition de stratégies, placement vis-à-vis de la concurrence, etc.). Des travaux proposant des solutions pour collecter des informations à partir de textes remontent aux premiers temps de la linguistique computationnelle, mais cette tâche a été bien délimitée depuis la fin des années 1980 à travers la définition du domaine de l'Extraction d'Information. Ce domaine a vu naître et évoluer de nombreuses techniques se fondant principalement sur des méthodes de Traitement Automatique des Langues, et a connu un accroissement des performances aboutissant au développement de systèmes réellement utilisables dans un cadre commercial ou institutionnel.

Les systèmes d'Extraction d'Information actuels produisent des résultats satisfaisants sur des textes rédigés en respectant les normes d'écriture de leur langue, mais leurs performances chutent lorsqu'il s'agit de traiter des documents dont la forme s'écarte de ces normes. Nous constatons en effet un manque d'efficacité des techniques usuelles d'Extraction d'Information sur des textes écrits en utilisant des règles d'écriture propres, c'est-à-dire s'écartant des normes usuelles, ou comportant de très nombreuses altérations de ces normes d'un point de vue lexical ou syntaxique. Ce constat est particulièrement vrai pour les Notes de Communication Orale dont les contraintes de rédaction (rapidité de la prise de note et limitation de la quantité d'écrits) sont à l'origine de déviances vis-à-vis des normes d'écriture. En conséquence, de tels textes sont peu exploitables par les systèmes usuels d'Extraction d'Information alors qu'il s'agit de textes possédant un fort contenu informatif. Afin d'apporter une solution au problème de l'extraction d'informations à partir de Notes de Communication, nous avons défini la méthode **MEGET** qui s'appuie sur l'élaboration d'une ontologie modélisant les informations à rechercher et leur représentations lexicales dans les corpus étudiés, ainsi que sur l'utilisation de cette ontologie comme base de connaissance d'un système d'extraction (**SYGET**).

Notre travail a d'abord consisté en une étude détaillée des textes que nous regroupons sous la dénomination de Notes de Communication Orale. Cette étude nous a permis de dégager un ensemble de caractéristiques linguistiques. Il s'agit de particularités orthographiques, typographiques, morphologiques et syntaxiques (utilisation fréquente d'abréviations, de logogrammes et d'expressions combinant caractères numériques et alphabétiques, présence de nombreuses fautes d'orthographe et de syntaxe, omission volontaire de mots, de signes de ponctuation et de diacritiques) inhérentes aux contraintes de rédaction de tels textes. La confrontation de ces caractéristiques avec les principales méthodes d'Extraction d'Information nous a montré que les problèmes que posent les Notes de Communication Orale aux systèmes d'extraction proviennent de l'inadéquation des techniques de Traitement Automatique des Langues avec les particularités lexicales et/ou syntaxiques de ce type de textes. La plupart de ces techniques sont en effet réalisées en suivant des modèles linguistiques prédéfinis et exprimés au moyen de grammaires, de lexiques ou encore de patrons lexico-syntaxiques. Ces modèles sont liés aux normes de la langue, soit parce qu'ils sont définis *a priori* en fonction de règles sur la langue, soit parce qu'ils sont observés ou extraits automatiquement de textes rédigés en conformité avec ces normes (corpus journalistiques, scientifiques, littéraires, etc.). Ainsi lorsque les textes dévient fortement de ces normes, comme c'est le cas pour les Notes de Communication Orale, les techniques de Traitement Automatique des Langues sont peu efficaces car les modèles utilisés ne correspondent pas avec les structures linguistiques présentes dans les textes. De plus l'absence de ressources linguistiques spécialisées (corpus annoté, lexique) pour les corpus de Notes de Communication Orale interdit de recourir à des méthodes d'apprentissage qui pourraient permettre de dégager des modèles linguistiques spécifiques à ce type de texte.

À la suite de cette étude, nous nous sommes tournés vers l'idée qu'une méthode fondée non plus sur une analyse du corpus lui-même mais sur une modélisation des connaissances contenues dans le corpus, pourrait nous permettre de nous abstraire des problèmes dus aux caractéristiques linguistiques des Notes de Communication Orale. En suivant cette idée nous avons élaboré la méthode **MEGET**, une méthode d'extraction fondée sur la construction d'une ontologie modélisant les connaissances contenues dans les textes et intéressantes du point de vue des informations à extraire, et sur son utilisation comme base de connaissance d'un système automatique d'extraction. L'ontologie est construite grâce à un travail coopératif avec des experts et représentée par un ensemble de règles formelles. Le processus de construction de l'ontologie se déroule en deux étapes : d'abord la modélisation des informations à extraire fondée sur la définition d'un ensemble de prédicats (*ontologie des besoins*) et ensuite l'élaboration d'une *ontologie des termes* conceptualisant les termes du corpus en relation avec les concepts de l'ontologie des besoins, réalisée à partir d'une terminologie issue de l'application d'un processus d'extraction de termes sur le corpus et de la consultation de documents externes aux textes (lexiques). L'unification des deux ontologies produit le modèle complet appelé *ontologie d'extraction*. L'extraction des informations à partir du texte est effectuée par le système **SYGET** qui repère dans le texte les différentes instances des concepts et des relations décrits dans l'ontologie d'extraction. Son architecture se compose de quatre modules exécutés successivement :

- Un module de prétraitements effectuant un ensemble de traitements visant à corriger certains problèmes typographiques et morphologiques récurrents dans les Notes de Communication Orale ;
- Un module de génération de la base de règles construisant la base de règles du système à partir de l'ensemble de règles formelles décrivant l'ontologie d'extraction ;
- Un module d'étiquetage utilisant les règles de la base pour marquer par des balises XML chaque instance d'un concept de l'ontologie d'extraction ainsi que les relations existant entre les instances de concepts identifiées dans le texte. Le résultat de l'exécution de ce module est un ensemble de fichiers au format XML ;
- Un module de recueil des informations identifiant toutes les instances des concepts à rechercher (concepts exprimant les informations à extraire) grâce aux balises et les renvoyant sous la forme de fichiers texte structurés.

Nous avons mené des expérimentations sur deux corpus de Notes de Communication Orale dans le domaine bancaire. Il s'agit de corpus écrits en français et issus de notes prises, pour l'un pendant des entretiens entre des banquiers et leurs clients et, pour l'autre, lors de conversations téléphoniques dans le cadre d'opérations de phoning. Les résultats obtenus ont été jugés très satisfaisants en termes de rappel et de précision par les analystes (banquiers, commerciaux) désirant exploiter le contenu informatif de ces textes, notamment en regard des résultats des méthodes d'extraction de connaissances utilisées auparavant. Malgré ces bons résultats, ceux-ci sont encore entachés d'incomplétudes (informations trouvées de manière partielle se traduisant par des instances de prédicats non complètes) dont une part non négligeable est imputable au système lui-même (le restant étant dû à l'absence de certaines informations dans les corpus). Nous avons également procédé à une expérience sur un corpus de courriers électroniques formé de textes de qualité linguistique standard mais dans lesquels les informations à rechercher ne sont pas toujours dans des phrases (informations présentes dans des titres, des listes d'items, etc.). Cette expérimentation a été réalisée afin d'évaluer la possibilité d'utiliser notre méthode pour d'autres type de textes que les Notes de Communication Orale.

Perspectives

Une première perspective concerne l'amélioration des résultats de **SYGET** grâce au recours à des méthodes d'apprentissage. L'idée serait d'utiliser tel quel le système afin d'étiqueter un échantillon de corpus à partir de l'ontologie élaborée pour traiter ce corpus, puis de faire valider cet échantillon par des experts du domaine afin d'éliminer les erreurs. Ensuite il s'agirait d'utiliser un processus d'apprentissage sur l'échantillon pour faire *apprendre* au système des schémas lexico-syntaxiques à partir des unités lexicales étiquetées dans l'échantillon. Les schémas appris permettraient au système d'étiqueter le reste du corpus plus efficacement [Soderland 1999] et d'augmenter la qualité des informations extraites.

Une autre perspective d'amélioration de **SYGET** serait de modifier le système afin qu'il puisse remplir directement une base de données XML à partir d'un corpus. Chaque document XML généré par le système serait stocké dans une base de données XML (Base de

données XML native ou NXD) dont la structure serait spécifiée par une DTD (Définition du Document Type) ou d'un *XML Schema* issu automatiquement de l'ontologie d'extraction. Les informations identifiées par le système seraient alors directement exploitables par des techniques d'analyse de base de données (requêtes, fouille de données, etc.) via un système de gestion de base de données conçu pour traiter des bases de données XML (SGBD natif XML). Une telle perspective est intéressante car elle permettrait au système de s'intégrer facilement dans des applications de Fouille de Données, de Fouille de Textes (cf. section 1.2.2) ou d'Extraction et de Gestion des Connaissances (EGC) [Hérin & Zighed 2002].

Sans avoir à effectuer la modification proposée dans le paragraphe précédent, l'intégration du système **SyGET** dans des applications de traitement et d'analyse documentaire ou d'Extraction et de Gestion des Connaissances est une des perspectives les plus intéressantes de notre travail. Les informations extraites par **SyGET** pourraient ainsi être utilisées pour améliorer les résultats d'un processus de recherche documentaire ou être exploitées dans des systèmes d'information et d'aide à la décision (SIAD) ou d'aide au management (Management Support Systems) [Marakas 2002].

Étendre le champ d'utilisation de **MeGET** à d'autres types de textes que les Notes de Communication Orale apparaît enfin comme une prolongation naturelle des travaux réalisés au cours de cette thèse. D'une part, il s'agirait de traiter des textes dont les caractéristiques linguistiques sont proches de celles des Notes de Communication Orale. Nous pensons particulièrement aux textes issus de messageries instantanées (MSN Messenger, Yahoo Messenger, etc.) qui se caractérisent par de nombreuses fautes orthographiques (fautes de frappe la plupart du temps), beaucoup d'abréviations et une syntaxe limitée [Anis 2003]. D'autre part, notre méthode pourrait aider au traitement de textes écrits dans des langues peu dotées en outils informatiques mais également en ressources linguistiques [Berment 2004], comme les langues africaines [Diki-Kidiri & Atibakwa Baboya 2003] [Enguehard & Mbodj 2004]. Ce manque de ressources rend très difficile la réalisation d'applications de Traitement Automatique de la Langue, et par extension d'Extraction d'Information. La perspective de recourir à **MeGET** pour traiter de telles langues est motivée par sa distance vis-à-vis de ressources et d'outils linguistiques.

Bibliographie

[Abdelali & al. 2003] A. Abdelali, J. Cowie, D. Farwell, B. Ogden et S. Helmreich, Cross-Language Information Retrieval using Ontology, *Actes de TALN 2003 (10ème conférence sur le Traitement Automatique des Langues Naturelles)*, Nancy, France, Tome 2, pp. 117-126, 2003.

[Abeillé 1991] A. Abeillé, Analyseurs syntaxiques du français, *Revue TAL (Traitement Automatique des Langues),* Vol. 32, n° 2, pp. 107-120, 1991.

[Abeillé 1993] A. Abeillé, Les nouvelles syntaxes : grammaire d'unification et analyse du français, Armand Collin, Paris, 1993.

[Abeillé & Candito 2000] A. Abeillé et M.-H. Candito, FTAG: A Lexicalized Tree Adjoining Grammar for French, *Tree Adjoining Grammars: Formalisms, Linguistic Analysis and Processing*, A. Abeillé et O. Rambow (réd.), CSLI Publications, pp. 305-330, 2000.

[Abeillé & Rambow 2000] A. Abeillé et O. Rambow, Tree Adjoining Grammar: An overview, *Tree Adjoining Grammars: Formalisms, Linguistic Analysis and Processing*, A. Abeillé et O. Rambow (réd.), CSLI Publications, pp. 1–68, 2000.

[Abney 1990] S. Abney, Rapid Incremental Parsing with Repair, *Proceedings of the 6th New OED Conference: Electronic Text Research*, University of Waterloo, Waterloo, Ontario, Canada, pp. 1-9, 1990.

[Alpha & al. 2001] S. Alpha, P. Dixon, C. Liao et C. Yang, Oracle at TREC 10: Filtering and Question-Answering, *Proceeding of the Tenth Text REtrieval Conference (TREC 2001)*, Gaithersburg, USA, NIST Special Publication 500-250, pp. 423-433, 2000.

[Andersen & al. 1992] P. M. Andersen, P. J. Hayes, A. K. Huettner, I. B. Nirenburg, L. M. Schmandt et S. P. Weinstein, Automatic Extraction of Facts from Press Releases to Generate News Stories, *Proceedings of the Third Conference on Applied Natural Language Processing*, pp. 170-177, 1992.

[Anis 2003] J. Anis, Communication électronique scripturale et formes langagières : chats et SMS, *Actes des Quatrièmes Rencontres Réseaux Humains / Réseaux Technologiques*, Université de Poitiers, France, `http://oav.univ-poitiers.fr/rhrt/2002/`, 2003.

[Appelt & Israel 1999] D. Appelt et D. Israel, Introduction to Information Extraction Technology, *Tutorial for the International Joint Conference on Artificial Intelligence (IJCAI-99)*, Stockholm, Sweden, 1999.

[Attardi & Burrini 2001] G. Attardi et C. Burrini, The PISAB Question Answering System, *Proceedings of the Ninth Text Retrieval Conference (TREC 9)*, Gaithersburg, USA, NIST Special Publication 500-249, pp. 446-451, 2001.

[Aussenac-Gilles & al. 1992] N. Aussenac-Gilles, J.P. Krivine et J. Sallantin, L'acquisition des connaissances pour les systèmes à base de connaissances, *Éditorial de la revue Intelligence Artificielle*, Vol. 6(1-2), pp. 7-18, 1992.

[Aussenac-Gilles & al. 2000a] N. Aussenac-Gilles, B. Biébow et S. Szulman, Corpus analysis for conceptual modelling, *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000)*, pp. 13-20, 2000.

[Aussenac-Gilles & al. 2000b] N. Aussenac-Gilles, B. Biébow et S. Szulman, Modélisation du domaine par une méthode fondée sur l'analyse de corpus, *Actes d'IC 2000 (Ingénierie des Connaissances)*, Toulouse, France, 2000.

[Aussenac-Gilles & Condamines 2001] N. Aussenac-Gilles et A. Condamines, Entre textes et ontologies formelles : les Bases de Connaissances Terminologiques, *Ingénierie et capitalisation des connaissances*, M.Zacklad et M.Grundstein (réd.), Hermès, Paris, pp. 153-176, 2001.

[Baeza-Yates & Ribeiro-Neto 1999] R. Baeza-Yates et B. Ribeiro-Neto, Modern Information Retrieval, Addison-Wesley Longman Publishing Co, 1999.

[Bachimont 2001] B. Bachimont, Modélisation linguistique et modélisation logique des ontologies : l'apport de l'ontologie formelle, *Actes d'IC 2001 (Ingénierie des Connaissances)*, Presses Universitaires de Grenoble, France, pp. 349-368, 2001.

[Bear & al. 1997] J. Bear, D. Israel, J. Petit et D. Martin, Using Information Extraction to Improve Document Retrieval, *Proceedings of the Sixth Text REtrieval Conference (TREC 6)*, Gaithersburg, USA, NIST Special Publication 500-240, pp. 367-378, 1997.

[Beaujean & Littré 2003] A. Beaujean et E. Littré, Le Petit Littré – Dictionnaire de la langue française abrégé du dictionnaire de Littré, Librairie Générale Française (L.G.F.), ISBN 225313116-4, 2003.

[Bédard & Maurais 1983] E. Bédard et J. Maurais (réd.), La Norme Linguistique, Conseil supérieur de la langue française de Québec, Le Robert, ISBN 255105243-2, 1983.

[Benayache & al. 2004] A. Benayache, C. Barry, B. Chaput et M.-H. Abel, Construction of application ontology for e-learning, *Proceedings of the World Conference on E-Learning in Corporate, Government Healthcare & Higher Education (E-Learn04)*, Washington, USA, 2004.

[Benveniste 1966] E. Benveniste, Formes nouvelles de la composition nominale, *Problèmes de linguistique générale*, Gallimard, Paris, pp. 163-173, 1966.

[Bérard-Dugourd & al. 1988] A. Bérard-Dugourd, J. Fargues, M.-C. Landau et J.-P. Rogala, Natural Language Analysis Using Conceptual Graphs, *Proceedings of the International Computer Science Conference'88*, Hong-Kong, pp. 265-272, 1988.

[Berment 2004] D. Berment, Méthodes pour informatiser des langues et des groupes de langues peu dotées, *Thèse de Doctorat (Spécialité Informatique)*, Université Grenoble 1, France, 2004.

[Bernaras & al. 1996] A. Bernaras, I. Laresgoiti et J. Corera, Building and reusing ontologies for electrical network applications, *Proceedings of ECAI96 (12th European Conference on Artificial Intelligence)*, Budapest, Hungary, pp. 298-302, 1996.

[Biber 1989] D. Biber, A typology of English texts, *Linguistics*, Vol. 27, pp. 3-43, 1989.

[Biber & Finegan 1986] D. Biber et E. Finegan, An Initial Typology of English Text Types, *Corpus Linguistics II: New Studies in the Analysis and Exploitation of Computer Corpora*, J. Aarts et W. Meijs (réd.), Editions Rodopi, Amsterdam, Vol. 47, pp. 19-46, 1986.

[Black 1997] W. J. Black, FACILE: Fined-Grained Multilingual Text Categorisation and Information Extraction, *Natural Language Processing: Extracting Information for Business Needs*, Unicom Seminars Ltd, London, Great-Britain, pp. 119-131, 1997.

[Blanquet 1997] M.-F. Blanquet, Science de l'information et philosophie : une communauté d'interrogations, Paris, ADBS Éditions, ISBN 284365001-1, 1997.

[Blum & al. 2005] C. Blum, J. Pruvost, K. Alaoui et G. Bady (collectif), Le nouveau Littré : Edition augmentée du Petit Littré, ISBN 284431249-7, 2005.

[Borst 1997] W. N. Borst, Construction of Engineering Ontologies, *PhD Thesis*, University of Twente, Netherlands, 1997.

[Borst & al. 1997] W. N. Borst, J. M. Akkermans, et J. L. Top, Engineering Ontologies, *International Journal of Human Computer Studies (Special Issue on Using Explicit Ontologies in KBS Development)*, Vol. 46, pp. 365-406, 1997.

[Bouaud & al. 1995] J. Bouaud, B. Bachimont, J. Charlet et P. Zweigenbaum, Methodological Principles for Structuring an Ontology, *Proceedings of IJCAI'95 Workshop on Basic Ontological Issues in Knowledge Sharing*, Montreal, Canada, ACM Press, 1995.

[Bourigault 1994] D. Bourigault, LEXTER, Un Logiciel d'Extraction de Terminologie. Application à l'acquisition de connaissances à partir de textes, *Thèse de Doctorat (Spécialité Mathématiques, informatique appliquée aux sciences de l'homme)*, École des Hautes Études en Sciences Sociales (EHESS), Paris, France, 1994.

[Bourigault & al. 2004] D. Bourigault, N. Aussenac-Gilles et J. Charlet, Construction de ressources terminologiques ou ontologiques à partir de textes : un cadre unificateur pour trois études de cas, *Revue d'Intelligence Artificielle*, Vol. 16, n° 1, pp. 87-110, 2004.

[Bourigault & Aussenac-Gilles 2003] D. Bourigault et N. Aussenac-Gilles, Construction d'ontologie à partir de textes, *Actes de TALN 2003 (10ème conférence sur le Traitement Automatique des Langues Naturelles)*, Batz-sur-Mer, France, Tome 2, pp. 27-49, 2003.

[Bourigault & Fabre 2000] D. Bourigault et C. Fabre, Approche linguistique pour l'analyse syntaxique de corpus, *Cahiers de Grammaire*, Université Toulouse – Le Mirail, n° 25, pp. 131-154, 2000.

[Bourigault & Jacquemin 2000] D. Bourigault et C. Jacquemin, Construction de ressources terminologiques, *Ingénierie des langues*, J.-M. Pierrel (réd.), Hermès, pp. 215-233, 2000.

[Bourigault & Slodzian 1999] D. Bourigault et M. Slodzian, Pour une terminologie textuelle, *Actes des troisièmes journées « Terminologie et Intelligence Artificielle » (TIA99)*, Nantes, France, *Terminologies nouvelles*, n° 19, pp. 29-32, 1999.

[Bozsak & al. 2002] E. Bozsak, M. Ehrig, S. Handschuh, A. Hotho, A. Maedche, B. Motik, D. Oberle, C. Schmitz, S. Staab, L. Stojanovic, N. Stojanovic, R. Studer, G. Stumme, Y. Sure, J. Tane, R. Volz et V. Zacharias, KAON - towards a large scale semantic web, *Proceedings of the Third International Conference on E-Commerce and Web Technologies (EC-Web 2002)*, Aix-en-Provence, France, Lecture Notes in Computer Science, LNCS 2455, Springer, pp. 304-313, 2002.

[Bresnan & Kaplan 1982] J. Bresnan et R. Kaplan, Lexical functional grammar: A formal system for grammatical representation, *The Mental Representation of Grammatical Relations*, Cambridge MIT Press, pp. 173-281, 1982.

[Brill 1993] E. Brill, A Corpus-Based Approach to Language Learning, *PhD thesis*, University of Pennsylvania, USA, 1993.

[Burger & al. 2001] J. Burger, C. Cardie, V. Chaudhri, R. Gaizauskas, S. Harabagiu, D. Israel, C. Jacquemin, C.-Y. Lin, S. Maiorano, G. Miller, D. Moldovan, B. Ogden, J. Prager, E. Riloff, A. Singhal, R. Shrihari, T. Strzalkowski, E. Voorhees et R. Weishedel, Issues, Tasks and Program Structures to Roadmap Research in Question & Answering (Q&A), *NIST DUC Road Mapping*, `http://www-nlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc`, 2001.

[Califf 1999] M. E. Califf, Relational Learning Techniques for Natural Language IE, *Ph.D. thesis*, Univ. Texas, Austin, 1999.

[Califf & Mooney 1999] M. E. Califf et R. J. Mooney, Relational Learning of Pattern-Match Rules for Information Extraction, *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, Orlando, USA, pp. 328-334, 1999.

[Cardie 1997] C. Cardie, Empirical Methods in Information Extraction, *AI Magazine*, Vol. 18(4), pp. 65-79, 1997.

[Chaput & al. 2004] B. Chaput, A. Benayache, C. Barry et M.-H. Abel, Une expérience de construction d'ontologie d'application pour indexer les ressources d'une formation en statistique, *Actes des Trente-sixièmes Journées Françaises de Statistique (SFDS'04)*, Montpellier, France, 2004.

[Charlet & al. 2000] J. Charlet, M. Zacklad, G. Kassel et D. Bourigault, *Ingénierie des connaissances, Evolutions récentes et nouveaux défis*, Eyrolles, ISBN 221209110-9, 2000.

[Charniak 1993] E. Charniak, Statistical Language Learning, MIT Press, Cambridge, ISBN 026253141-0, 1993.

[Chinchor 1997] N. Chinchor, MUC-7 Named Entity Task Definition, *Proceedings of the Seventh Message Understanding Conference*, `http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/ne_task.html`, 1997.

[Chinchor & Marsh 1998] N. Chinchor et E. Marsh, MUC-7 Information Extraction Task Definition, *Proceedings of the Seventh Message Understanding Conference*, `http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/ie_task.html`, 1998.

[Chomsky 1957] N. Chomsky, Syntactic structures, Mouton de Gruyter (réd.), The Hague, ISBN 311015412-9, 1957.

[Church 1988] K. W. Church, A stochastic parts program and noun phrase parser for unrestricted texts, *Proceedings of the Second Conference on Applied Natural Language Processing*, Austin, Texas, USA, 1988.

[Church & Hanks 1990] K. W. Church et P. Hanks, Word association norms, mutual information and lexicography, *Computational Linguistics*, Vol. 16(1), pp. 22-29, 1990.

[Ciravegna 2001] F. Ciravegna, Adaptive Information Extraction from Text by Rule Induction and Generalisation, *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI'2001)*, Seattle, USA, pp. 1251-1256, 2001.

[Cooper & Rüger 2000] R. J. Cooper et S. M. Rüger, A Simple Question Answering System, *Proceedings of the Ninth Text REtrieval Conference (TREC 9)*, Gaithersburg, USA, NIST Special Publication 500-249, 2000.

[Cowie 1983] J. Cowie, Automatic Analysis of descriptive texts, *Proceedings of the ACL Conference on Applied Language Processing*, 1983.

[Cunningham 1996] H. Cunningham, AVENTINUS, ECRAN and GATE -- Information Extraction and Language Engineering, *Proceedings of the Workshop on Multilingual Access to textual Information (MAIN-96)*, Volterra, Italy, 1996.

[Cunningham 1999] H. Cunningham, Information Extraction – A User Guide (updated version), *Research Memorandum CS--99--07*, Department of Computer Science, University of Sheffield, 1999.

[Cunningham 2002] H. Cunningham, GATE, a general architecture for text engineering, *Computers and the Humanities*, Vol. 36, pp. 223-254, 2002.

[Cunningham & al. 2003] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, C. Ursu et M. Dimtrov, Developing Language Processing Components with GATE (a User Guide), *Technical Report*, Department of Computer Science, University of Sheffield, 2003.

[Cutting & al. 1992] D. Cutting, J. Kupiec, J. Pedersen et P. Sibun, A practical part-of-speech tagger, *Proceedings of the Third Conference on Applied Language Processing (ANLP-92)*, Trente, Italy, 1992.

[Daille 1994] B. Daille, Approche mixte pour l'extraction de terminologie : statistique lexicale et filtres linguistiques, *Thèse de Doctorat (Spécialité Informatique)*, Université Paris VII, France, 1994.

[Daille 1996] B. Daille, Study and Implementation of Combined Techniques for Automatic Extraction of Terminology, *The Balancing Act: Combining Symbolic and Statistical Approaches to Language*, P. Resnik et J. Klavans (réd.), Cambridge, MA, MIT Press, pp. 49-66, 1996.

[Daille 2001] B. Daille, Qualitative terminology extraction, *Recent Advances in Computational Terminology*, D. Bourigault, C. Jacquemin et M.-C. L'Homme (réd.), John Benjamins, pp. 149-166, 2001.

[Daille 2003] B. Daille, Conceptual structuring through term variations, *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, Sapporo, Japon, pp. 9-16, 2003.

[David & Plante 1990] S. David et P. Plante, De la nécessité d'une approche morpho-syntaxique dans l'analyse de textes, *ICO*, Vol. 2(3), pp.140-154, 1990.

[Declerk & al. 2001] T. Declerck, P. Wittenburg et H. Cunningham, The Automatic Generation of Formal Annotations in a Multimedia Indexing and Searching Environment, *Proceedings of the ACL/EACL Workshop on Human Language Technology and Knowledge Management*, Toulouse, France, pp. 129-136, 2001.

[De Bertrand de Beuvron 1992] F. De Bertrand de Beuvron, Un système de programmation logique pour la création d'interfaces Homme-Machine en langue naturelle, *Thèse de Doctorat (Spécialité Informatique)*, Université de Technologie de Compiègne, France, 1992.

[De Jong 1982] G. De Jong, An Overview of the FRUMP System, *Strategies for Natural Language Processing*, W. Lehnert et M. H. Ringle (réd.), Lawrence Erlbaum Associates, pp. 149-176, 1982.

[De Jong & Westerveld 2001] F. De Jong et T. Westerveld, MUMIS: multimedia indexing and searching, *Proceedings of Content-Based Multimedia Indexing (CBMI 2001)*, Brescia, Italy, pp. 423-425, 2001.

[De Villiers 1988] M.-E De Villiers, Multidictionnaire des difficultés de la langue française, Éditions Québec/Amérique, Montréal, Canada, ISBN 289037402-5, 1988.

[Diekema & al. 2002] A. R. Diekema, J. Chen, N. McCracken, N. E. Ozgencil, M. D. Taffet, O. Yilmazel et E. D. Liddy, Question Answering: CNLP at the TREC-2002 Question Answering Track, *Proceedings of the Eleventh Text Retrieval Conference (TREC 2002)*, Gaithersburg, USA, NIST Special Publication 500-251, 2002.

[Dieng-Kuntz & al. 2001] R. Dieng-Kuntz, O. Corby, F. Gandon, A. Giboin, J. Golebiowska, N. Matta et M. Ribière, Méthodes et outils pour la gestion des connaissances : une approche pluridisciplinaire du Knowledge Management, *Collection Informatiques (Série Systèmes d'information)*, Dunod, France, ISBN 210006300-6, 2001.

[Diki-Kidiri & Atibakwa Baboya 2003] M. Diki-Kidiri et E. Atibakwa Baboya, Les langues africaines sur la toile, *Les cahiers du RIFAL - Le traitement informatique des langues africaines*, n° 23, pp. 5-32, 2003.

[Drãgãnescu 1996] Mihai Drãgãnescu, L'Universalité Ontologique de l'Information (adaptation en français par Yves Kodratoff), Editura Academiei Române, `http://www.racai.ro/books/draganescu/tdm.html`, 1996.

[Dubreil 2002] E. Dubreil, Une déviance communicationnelle ? Étude de cas, *Mémoire de DEA (Spécialité Science du Langage)*, Université de Nantes, 2002.

[Dunning 1993] T. Dunning, Accurate Methods for the Statistics of Surprise and Coincidence, *Computational Linguistics*, Vol. 19, n° 1, pp. 61-74, 1993

[Eikvil 1999] L. Eikvil, Information Extraction from World Wide Web - A Survey, *Technical Report n° 945*, ISBN 825390429-0, 1999.

[El-Bèze 1993] M. El-Bèze, Les Modèles de Language Probabilistes : Quelques Domaines d'Applications, *Habilitation à diriger la recherche*, LIPN, Université Paris-Nord, 1993.

[Enguehard 1992] C. Enguehard, Acquisition naturelle automatique d'un réseau sémantique, *Thèse de Doctorat (Spécialité Informatique)*, Université de Technologie de Compiègne, France, 1992.

[Enguehard 1993] C. Enguehard, Acquisition de terminologie à partir de gros corpus, *Informatique & Langue Naturelle (ILN'93)*, Nantes, France, pp. 373-384, 1993.

[Enguehard 2000] C. Enguehard, Flexible-equality of terms: definition and evaluation, *Proceedings of the International Conference on Flexible Query Answering Systems*, Henrik L. Larsen, Janusz Kacprzyk, Slawonir Zadrozny, Troels Andreasen, et Henning Christiansen (réd.), ISBN 379081347-8, pp. 289-300, 2000.

[Enguehard & Mbodj 2004] C. Enguehard et C. Mbodj, Des correcteurs orthographiques pour les langues africaines, *La correction automatique : bilan et perspectives*, *BUlletin de Linguistique Appliquée et Générale (BULAG)*, n° 29, pp. 51-68, 2004.

[Enguehard & Pantéra 1995] C. Enguehard et L. Pantéra, Automatic Natural Acquisition of a Terminology, *Journal of quantitative linguistics*, Vol. 2, n° 1, pp. 27-32, 1995.

[Even 2004] F. Even, Corpus Analysis to Extract Information, *Proceedings of the K-CAP 2003 (Second International Conference on Knowledge Capture) Workshop on Knowledge Markup and Semantic Annotation (SemAnnot 2003)*, Floride, USA, 2003, *CEUR Workshop Proceedings*, ISSN 1613-0073, Vol. 101, pp. 95-100, 2004.

[Even & Enguehard 2002] F. Even et C. Enguehard, Extraction d'informations à partir de corpus dégradés, *Actes de TALN 2002 ($9^{ème}$ conférence sur le Traitement Automatique des Langues Naturelles)*, Tome 1, Nancy, France, pp. 105-114, 2002.

[Even & Enguehard 2003] F. Even et C. Enguehard, Specific Domain Model Building for Information Extraction from poor quality corpus, *Proceedings of the EUROLAN'03 International Workshop on Ontologies and Information Extraction*, Bucharest, Romania, pp. 3-9, 2003.

[Faloutsos & Oard 1996] C. Faloustos et D. Oard, A survey of information retrieval and filtering methods, Technical Report n° 3514, University of Maryland, College Park, 1996.

[Fayyad & al. 1996] U.M. Fayyad, G. Piatetsky-Shapiro et P. Smyth, From Data Mining to Knowledge Discovery: An Overview, *Advances in Knowledge Discovery and Data-Mining*, AAAI/MIT Press, pp. 1-36, 1996.

[Felber 1987] H. Felber, Manuel de Terminologie, Unesco, Paris, 1987.

[Feldman & al. 1998] R. Feldman, M. Fresko, Y. Kinar, Y. Lindell, O. Liphstat, M. Rajman, Y. Schler et O. Zamir, Text Mining at the Term Level, *Proceedings of the 2nd European Symposium on Principles of Data Mining and Knowledge Discovery*, Nantes, France, pp. 65-73, 1998.

[Felbaum 1998] C. Felbaum (réd.), Wordnet: An Electronical Database, Cambridge, MIT Press, 1998.

[Ferret & al. 1999] O. Ferret, B. Grau, G. Illouz, C. Jacquemin et N. Masson, QALC - the question-answering program of the language and cognition group at LIMSI-CNRS, *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*, Gaithersburg, USA, NIST Special Publication 500-246, E. M. Voorhees et D. K. Harman (réd.), 1999.

[Ferret & al. 2002a] O. Ferret, B. Grau, M. Hurault-Plantet, G. Illouz, C. Jacquemin, L. Monceaux, I. Robba et A.Vilnat, How NLP Can Improve Question Answering, *Revue Knowledge Organization*, Vol. 29, n° 3-4, pp. 135-155, 2002.

[Ferret & al. 2002b] O. Ferret, B. Grau, M. Hurault-Plantet, G. Illouz, L. Monceaux, I. Robba et A. Vilnat, Recherche de la réponse fondée sur la reconnaissance du focus de la question, *Actes de TALN 2002 ($9^{ème}$ conférence sur le Traitement Automatique des Langues Naturelles)*, Nancy, France, Tome 1, pp. 307-316, 2002.

[Foulquié 1962] P. Foulquié, Dictionnaire de la langue philosophique, Presse Universitaire de France, Paris, France, 1962.

[Fourour 2004] N. Fourour, Identification et catégorisation des entités nommées dans les textes français, *Thèse de Doctorat (Spécialité Informatique)*, LINA, Université de Nantes, France, 2004.

[Fox 1992] M. S. Fox, The TOVE Project: A Common-sense Model of the Enterprise, *Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, F. Belli et F. J. Radermacher (réd.), Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence subseries), LNAI 604, Springer, pp. 25-34, 1992.

[Frath 1997] P. Frath, Sémantique, référence et acquisition automatique de connaissances à partir de textes, *Thèse de doctorat (Spécialité Sciences du Language)*, Université Marc Bloch, Strasbourg, France, 1997.

[Frath & al. 2000] P. Frath, R. Oueslati et F. Rousselot, Identification de relations sémantiques par repérage et analyse de cooccurrences de signes linguistiques, *Ingénierie des connaissances. Evolutions récentes et nouveaux défis*, J. Charlet, M. Zacklad, G. Kassel et D. Bourigault (réd.), Eyrolles, Paris, 2000.

[Frawley & al. 1991] W.J. Frawley, G. Piatetsky-Shapiro et C.J. Matheus, Knowledge Discovery in Databases: An Overview, *Knowledge Discovery in Databases*, MIT Press, pp. 1-27, 1991.

[Fürst 2004] F. Fürst, Contribution à l'ingénierie des ontologies : une méthode et un outil d'opérationalisation, *Thèse de Doctorat (Spécialité Informatique)*, LINA, Université de Nantes, France, 2004.

[Gadet 1997] F. Gadet, Le français ordinaire, 2ème édition, Armand Colin/Masson, Paris, ISBN 220001615-8, 1997.

[Gaizauskas 1995] R. Gaizauskas, XI: A knowledge representation language based on cross-classification and inheritance, *Research Memorandum CS-95-24*, Department of Computer Science, University of Sheffield, 1995.

[Gaizauskas 2002] R. Gaizauskas, An Information Extraction Perspective on Text Mining: Tasks, Technologies and Prototype Applications, *Euromap Text mining Seminar*, Londres, Grande-Bretagne, 2002.

[Gaizauskas & al. 1995] R. Gaizauskas, T. Wakao, K. Humphreys, H. Cunningham et Y. Wilks, University of Sheffield: Description of the LaSIE System as used for MUC-6, *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, Morgan Kaufmann, pp. 207-220, 1995.

[Gaizauskas & Wilks 1998] R. Gaizauskas et Y. Wilks, Information Extraction: Beyond Document Retrieval, *Computational Linguistics and Chinese Language Processing*, Vol. 3, n° 2, pp. 17-60, 1998.

[Gallager 1968] R. G. Gallager, Information Theory and Reliable Communication, Wiley Text Books, ISBN 047129048-3, 1968.

[Gaumer 2002] G. Gaumer, Résumé de données en Extraction de Connaissances à partir des Données (ECD) Application aux données relationnelles et textuelles, *Thèse de Doctorat (Spécialité Informatique)*, IRIN, Université de Nantes, France, 2002.

[Glasgow & al. 1997] B. Glasgow, A. Mandell, D. Binney, L. Ghemri et D. Fisher, MITA: An Information Extraction Approach to Analysis of Free-form Text in Life Insurance Applications, *Proceeding of the Ninth Conference on Innovative Applications of Artificial Intelligence (AAAI-97)*, Providence, Rhode Island, AAAI/MIT Press, pp. 992-999, 1997.

[Golebiowska 2002] J. Golebiowska, Exploitation des ontologies pour la mémoire d'un projet-véhicule : Méthode et outil SAMOVAR, *Thèse de Doctorat (Spécialité Informatique)*, INRIA, Université de Nice-Sophia Antipolis, France, 2002.

[Gómez-Pérez & al. 1996] A. Gómez-Pérez, M. Fernández et A. J. de Vicente, Towards a Method to Conceptualize Domain Ontologies, *Proceedings of the European Conference on Artificial Intelligence (ECAI'96)*, pp. 41-52, 1996.

[Gómez-Pérez & al. 2004] A. Gómez-Pérez, M. Fernández-López et O. Corcho, Ontological Engineering, *Advanced Information and Knowledge Processing (Series)*, Springer-Verlag, ISBN 185233551-3, 2004.

[Grau 2002] B. Grau, EQueR : Évaluation de systèmes de Questions Réponses, *Project definition, Part of the EVALDA evaluation initiative*, LIMSI-CNRS, Orsay, France, 2002.

[Grevisse 1997] M. Grevisse, Le Bon Usage -- Grammaire française, André Goosse (réd.), 13e édition, 1993-1997, DeBoeck-Duculot, Paris, ISBN 280111045-0, 1997.

[Grishman 1996] R. Grishman, TIPSTER Architecture Design Document Version 2.2 (Tinman Architecture), *Technical Report*, DARPA, 1996.

[Grishman 1997] R. Grishman, Information Extraction: Techniques and Challenges, *Information Extraction (a multidisciplinary approach to an emerging technology)*, M. T. Pazienza (réd.), Lectures Notes in Computer Science (Lecture Notes in Artificial Intelligence subseries), LNAI 1299, Springer, pp. 10-27, 1997.

[Grosz & al. 1995] B. J. Grosz, A. K. Joshi et S. Weinstein, Centering: a Framework for Modeling the Local Coherence of Discourse, *Computational Linguistics*, Vol. 21(2), pp. 203-225, 1995.

[Gruber 1993] T. Gruber, A translation approach to portable ontology specifications, *Knowledge Acquisition*, Vol. 5 (2), pp. 199-220, 1993.

[Guarino 1997] N. Guarino, Understanding, building and using ontologies, *International Journal of Human and Computer Studies*, Vol. 46(2/3), pp. 293-310, 1997.

[Guarino 1998] N. Guarino, Formal ontology and information systems, *Proceedings of the First International Conference on Formal Ontologies in Information Systems (FOIS)*, Trente, Italy, IOS-Press, pp. 3-15, 1998.

[Guarino & Giaretta 1995] N. Guarino et P. Giaretta, Ontologies and knowledge bases towards a terminological clarification, *Proceedings of KB&KS'95*, Amsterdam, Pays-Bas, IOS Press, pp. 25-32, 1995.

[Guarino & Welty 2000] N. Guarino et W. Welty, A Formal Ontology of Properties, *Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management (KAW'99)*, Banff, Canada, LNCS 1937, pp. 97-112, 2000.

[Guilbert & al. 1986] L. Guilbert, R. Lagane et G. Niobey (réd.), Grand Larousse de la langue française, Edition Larousse, Paris, 1986.

[Hachey & al. 2003] B. Hachey, C. Grover, V. Karkaletsis, A. Valarakos, M. T. Pazienza, M. Vindigni, E. Cartier et J. Coch, Use of Ontologies for Cross-lingual Information Management in the Web, *Proceedings of the Ontologies and Information Extraction International Workshop held as part of the EUROLAN 2003*, Bucharest, Romania, 2003.

[Hahn & Romacker 2000] U. Hahn et M. Romacker, Content management in the SynDiKATe system: How technical documents are automatically transformed to text knowledge bases, *Data & Knowledge Engineering*, Vol. 35 (2), pp. 137-159, 2000.

[Harabagiu & al. 2001] S. Harabagiu, D. Moldovan, M. Pasca, M. Surdeanu, R. Mihalcea, R. Girju, V. Rus, F. Lacatusu, P. Morarescu & R. Bunescu, Answering complex, list and context questions with LCC's Question-Answering Server, *Proceedings of the Tenth Text Retrieval Conference (TREC 2001)*, Gaithersburg, USA, NIST Special Publication 500-250, pp. 355-361, 2001.

[Harris 1951] Z. S. Harris, Structural Linguistics, University of Chicago Press, 1951.

[Hérin & Zighed 2002] D. Hérin et D. Zighed (réd.), EGC 2002 Extraction et gestion des connaissances, Extraction des connaissances et apprentissage, Hermès, Vol. 1, n° 4, 2002.

[Hirschman & Chinchor 1997] L. Hirschman et N. Chinchor, MUC-7 Coreference Task Definition, *Proceedings of the Seventh Message Understanding Conference*, `http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/co_task.html`, 1997.

[Hirst & Budanitsky 2004] G. Hirst, A. Budanitsky, Correcting real-word spelling errors by restoring lexical cohesion, *Natural Language Engineering*, to appear.

[Hobbs & al. 1992] J. R. Hobbs, D. Appelt, J. Bear, M. Tyson et D. Magerman, Robust Processing of Real-Word Natural-Language Texts, *Text-Based Intelligent Systems: Current Research and Practise in Information Extraction and Retrieval*, Laurence Erlbaum Associates, pp. 13-23, 1992.

[Hobbs & al. 1996] J. R. Hobbs, D. Appelt, J. Bear, D. Israel, M. Kameyama, M. Stickel et M. Tyson, FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text, *Finite State Devices for Natural Language Processing*, Roche & Schabes réd., MIT Press, Cambridge MA, pp. 383-406, 1996.

[Holowczak & Adam 1997] R. D. Holowczak et N. R. Adam, Information Extraction based Multiple-Category Document Classification for the Global Legal Information Network, *Proceedings of the Ninth Conference on Innovative Applications of Artificial Intelligence*, AAAI/MIT Press, pp. 1013-1018, 1997.

[Houben 2004] F. Houben, Mot vide, mot plein ? Comment trancher localement, *Actes de RECITAL 2004 (Rencontre des Etudiants Chercheurs en Informatique pour le Traitement Automatique des Langues Naturelles)*, Fès, Maroc, 2004.

[Hovy & al. 2001] E. Hovy, U. Hermjakob et C-Y Lin, The Use of External Knowledge of Factoid QA, *Proceedings of the Tenth Text REtrieval Conference (TREC 2001)*, Gaithersburg, USA, NIST Special Publication 500-250, pp. 644-652, 2001.

[Huffman 1995] S. B. Huffman, Learning information extraction patterns from examples, *Proceedings of the IJCAI'95 Workshop on New Approaches to Learning for natural Language Processing*, Montréal, Canada, pp. 127-133, 1995.

[Ichimura & al. 2001] Y. Ichimura, Y. Nakayama, M. Miyoshi, T. Akahane, T. Sekiguchi et Y. Fujiwara, Text Mining System for Analysis of A Salesperson's Daily Report, *Proceedings of PACLING'01 (Pacific Association for Computational LINGuistics)*, Kitakyushu, Japon, 2001.

[ISO 1969] ISO R 1087, *Vocabulaire de la terminologie*, Organisation internationale de normalisation, Genève, Suisse, 1969.

[Jackendoff 1977] R. Jackendoff, *X-bar syntax: A Study of Phrase Structure*, MIT Press, Cambridge, 1977.

[Jacobs & Rau 1990] P. S. Jacobs et L. F. Rau, SCISOR : Extracting Information from on-line news, *Communications of the ACM*, Vol. 33, n° 11, pp. 88-97, 1990.

[Jacquemin 1991] C. Jacquemin, Transformations des noms composés. *Thèse de Doctorat (Spécialité Informatique)*, Université de Paris 7, Paris, France, 1991.

[Joshi 1987] A. Joshi, Introduction to Tree Adjoining Grammar, *The Mathematics of Language*, A. Manaster-Ramer (réd.), Amsterdam/Philadelphia, 1987.

[Joshi & Schabes 1992] A. Joshi et Y. Schabes, Tree-adjoining grammar and lexicalized grammars, *Tree automata and languages*, M. Nivat et A. Podelski (réd.), North-Holland, ISBN 044489026-2, 1992.

[Justeson & Katz 1995] J. Justeson et S. Katz, Technical terminology: some linguistic properties and an algorithm for identification in text, *Natural Language Engineering*, Vol. 1(1), pp. 9-27, 1995.

[Kassel 2002] G. Kassel, OntoSpec : une méthode de spécification semi-informelle d'ontologies, *Actes des 13èmes Journées Francophones d'Ingénierie des Connaissances (IC'2002)*, Rouen, France, pp. 75-87, 2002.

[Kayser 1997] D. Kayser, La représentation des connaissances, *Collection Informatique*, Hermès Sciences Publications, ISBN 286601647-5, 1997.

[Kim & al. 1999] H. M. Kim, M. S. Fox et M. Gruninger, An ontology for quality management - enabling quality problem identification and tracing, *BT Technology Journal*, Vol. 17, n° 4, pp. 131-140, 1999.

[Lame 2000] G. Lame, Knowledge acquisition from texts towards an ontology of French law, *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000)*, pp. 53-62, 2000.

[Lebart & Salem 1994] L. Lebart et A. Salem, Statistique Textuelle, Dunod, ISBN 210002239-3, 1994.

[Lecomte & Naït-Abdallah 2003] A. Lecomte et A. Naït-Abdallah, Un modèle de raisonnement avec propositions implicites, *Actes des Journées Nationales sur les Modèles de Raisonnement (JNMR'03)*, Institut Henri Poincaré, Paris, France, pp. 201-217, 2003.

[Lecomte & Paroubek 1996] J. Lecomte et P. Paroubek, Le catégoriseur d'Eric Brill. Mise en oeuvre de la version entraînée à l'InaLF, *Rapport interne*, CNRS-INaLF, Nancy, 1996.

[Lefébure & Venturi 2005] R. Lefébure et G. Venturi, *Gestion de la relation client*, Eyrolles, ISBN 221211331-5, 2005.

[Lehnert 1978] W. Lehnert, The Process of Question Answering – A Computer Simulation of Cognition, *Lawrence Erlbaum Associates*, Hillsdale, New Jersey, USA, 1978.

[Lehnert & al. 1992] W. Lehnert, C. Cardie, D. Fisher, J. McCarthy, E. Riloff, S. Soderland, University of Massachusetts: Description of the CIRCUS System as Used for MUC-4, *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, pp. 151-158, 1992.

[Lehnert & al. 1994] W. Lehnert, C. Cardie, D. Fisher, J. McCarthy, E. Riloff et S. Soderland (1994), Evaluating an Information Extraction System, *Journal of Integrated Computer-Aided Engineering*, Vol. 1(6), pp. 453-472, 1994.

[Lin & Katz 2003] J. Lin et B. Katz, Question Answering techniques for the World Wid Web, *Tutoriel of the European ACL (EACL 2003)*, Budapest, Hungary, 2003.

[Lopez 1999] P. Lopez, Analyse d'énoncés oraux pour le Dialogue Homme-Machine à l'aide de Grammaires Lexicalisées d'Arbres, *Thèse de doctorat (Spécialité Informatique)*, Université Henri Poincaré-Nancy 1, France, 1999.

[Lytinen & Gershman 1986] S. L. Lytinen et A. Gershman, ATRANS: Automatic processing of money transfert messages, *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86)*, Philadelphia, USA, pp. 1089-1093, 1986.

[Maedche & al. 2002] A. Maedche, G. Neumann et S. Staab, Bootstrapping an Ontology Based Information Extraction System, *Intelligent Exploration of the Web*, P.S. Szczepaniak, J. Segovia, J. Kacprzyk, L.A. Zadeh (réd.), Studies in Fuzziness and Soft Computing, J. Kacprzyk (réd.), Springer-Verlag, pp. 345-359, 2002.

[Marakas 2002] G. M. Marakas, Decision Support Systems and Megaputer, Prentice-Hall, ISBN 013101879-6, 2002.

[Marcus & al. 1993] M. P. Marcus, B. Santorini et M. A. Marcinkiewicz, Building a Large Annotated Corpus of English: The penn Treebank, *Computational Linguistics*, Vol. 19(2), pp. 313-330, 1993.

[Minel 2002] J.-L. Minel, Filtrage sémantique. Du résumé automatique à la fouille de textes, Hermès, Paris, ISBN 274620602-1, 2002.

[Mizoguchi 1997] R. Mizoguchi et M. Ikeda, Towards ontology engineering, *Proceedings of the Joint Pacific Asian Conference on Expert System (PACES'97)*, Singapore, pp. 259-266, 1997.

[Moldovan & al. 1999] D. Moldovan, S. Harabagiu, M. Pasça, R. Mihalcea, R. Goodrum, R. Gîrju et V. Rus, LASSO: A Tool for Surfing the Answer Net, *Proceedings of the Eight Text Retrieval Conference (TREC-8)*, Gaithersburg, USA, pp. 175-184, 1999.

[Monceaux 2002] L. Monceaux, Adaptation du niveau d'analyse des interventions dans un dialogue Application à un système de question – réponse, *Thèse de Doctorat (Spécialité Informatique)*, LIMSI, CNRS, Université de Paris XI, Orsay, France, 2002.

[Monceaux & Robba 2002] L. Monceaux et I. Robba, Les analyseurs syntaxiques : atouts pour une analyse des questions, *Actes de TALN 2002 ($9^{ème}$ conférence sur le Traitement Automatique des Langues Naturelles)*, Nancy, France, Tome 1, pp. 195-204, 2002.

[Morin 1999a] E. Morin, Extraction de liens sémantiques entre termes à partir de corpus de termes techniques, *Thèse de Doctorat (Spécialité Informatique)*, IRIN, Université de Nantes, France, 1999.

[Morin 1999b] E. Morin, Des patrons lexico-syntaxiques pour aider au dépouillement terminologique, *Revue TAL (Traitement Automatique des Langues)*, Vol. 40, n°1, pp. 143-166, 1999.

[MUC-7 1998] MUC-7, *Proceedings of the Seventh Message Understanding Conference*, Fairfax, Virginie, USA, `http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/muc_7_toc.html`, 1998.

[Muslea & al. 1998] I. Muslea, S. Minton et C. Knoblock, STALKER : Learning Extraction Rules for Semistructured, Web-based Information Sources, *Workshop on AI and Information Integration in conjunction with the 15th National Conference on Artificial Intelligence (AAAI-98)*, Madison, Wisconsin, USA, pp. 74-81, 1998.

[Naur & al. 1960] P. Naur (réd.), J. W. Backus, F. L. Bauer, J. Green, C. Katz, J. McCarthy, A. J. Perlis, H. Rutishauser, K. Samelson, B. Vauquois, J. H. Wegstein, A. van Wijngaarden et M. Woodger, Report on the Algorithmic Language ALGOL 60, *Communication of the ACM*, ACM Press, Vol. 3, n° 5, pp. 299-314, 1960.

[Niles & Pease 2001] I. Niles et A. Pease, Towards a standard upper ontology, *Proceedings of the international conference on Formal Ontology in Information Systems (FOIS)*, Ogunquit, USA, ACM Press, Vol. 2001, pp. 3-9, 2001.

[Nobécourt 2000] J. Nobécourt, A method to build formal ontologies from texts, *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000)*, Juan-les-Pins, France, pp. 21-27, 2000.

[Oueslati 1999] R. Oueslati, Aide à l'acquisition de connaissances à partir de corpus, *Thèse de Doctorat (Spécialité Informatique)*, Université Louis Pasteur, Strasbourg, France, 1999.

[Paquette 1983] J.-M. Paquette, Procès de normalisation et niveaux/registre de langue, *La Norme Linguistique*, Chapitre XIII, E. Bédard & J. Maurais (réd.), 1983.

[Paquette 2002] G. Paquette, Modélisation des connaissances et des compétences : Pour concevoir et apprendre, Presses de l'Université du Québec, 2002.

[Pérennou & al. 1986] G. Pérennou, P. Daubèze et F. Lahens, La vérification et la correction automatique de textes : le système VORTEX, *Technique et Science Informatique*, Vol. 5, n° 4, pp. 285-305, 1986.

[Piazenza 1997] M. T. Pazienza (réd.), Information Extraction (a multidisciplinary approach to an emerging technology), *Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence subseries)*, LNAI 1299, Springer, pp. 10-27, 1997.

[Pietrosanti 1997] E. Pietrosanti et B. Graziado, Artificial Intelligence and Legal Text Management: Tools and Techniques for Intelligent Document Processing and Retrieval, *Natural Language Processing: Extracting Information for Business Needs*, Unicom Seminars Ltd., London, Great-Britain, pp. 277-291, 1997.

[Poibeau 2002] T. Poibeau, Extraction d'Information à base de connaissances hybrides, *Thèse de Doctorat (Spécialité Informatique)*, LIPN, Université Paris-Nord, France, 2002.

[Poibeau & Balvet 2001] T. Poibeau et A. Balvet, Corpus-based lexical acquisition for Information Extraction, *Proceedings of IJCAI-2001 Workshop on Adaptive Text Extraction and Mining*, Seattle, USA, 2001.

[Poibeau & Dutoit 2002] T. Poibeau et D. Dutoit, Generating Extraction Patterns from a Large Semantic Network and an Untagged Corpus, *Proceedings of the SemaNet Workshop during the 19th International Conference on Computational Linguistics (COLING 2002)*, Taipei, Taiwan, 2002.

[Poibeau & Nazarenko 1999] T. Poibeau et A. Nazarenko. L'extraction d'information, une nouvelle conception de la compréhension de texte ? *Revue TAL (Traitement Automatique des Langues)*, Vol. 40, n° 2, pp. 87-115, 1999.

[Poesio & al. 2000] M. Poesio, H. Cheng, R. Henschel, J. Hitzeman, R. Kibble et R. Stevenson, Specifying the Parameters of Centering Theory: a Corpus-Based Evaluation using Text from Application-Oriented Domains, *Proceedings of the 38th Meeting of the Association for Computational Linguistics (ACL'2000)*, Hong-Kong, 2000.

[Pollard & Sag 1987] C. Pollard et I. Sag, Information-based Syntax end Semantics, CSLI Series, University of Chicago Press, 1987.

[Purnelle & al. 2004] G. Purnelle, C. Fairon et A. Dister (réd.), Le poids des mots, *Actes des 7èmes Journées internationales d'Analyse statistique des Données Textuelles*, Presses universitaires de Louvain, 2004.

[Rajman 1995] M. Rajman, Approche probabiliste de l'analyse syntaxique, *Revue TAL (Traitement Automatique des Langues)*, Vol. 36, n° 1-2, pp. 157-199, 1995.

[Rajman & Besançon 1997] M. Rajman et R. Besançon, Text Mining: Natural Language techniques and Text Mining applications, *Proceedings of the Seventh IFIP 2.6 Working Conference on Database Semantics (DS-7)*, Leysin, Suisse, pp. 50-65, 1997.

[Rajman & Besançon 1998] M. Rajman et R. Besançon, Text Mining – Knowledge extraction from unstructured textual data, *Proceedings of the Sixth Conference of International Federation of Classication Societies (IFCS-98)*, Rome, Italie, pp. 473-480, 1998.

[Rastier 1995] F. Rastier, Le terme : entre ontologie et linguistique, *Actes des 1ères journées « Terminologie et Intelligence Artificielle »*, Villetaneuse, *La banque des mots*, Numéro spécial 7-1995, pp. 35-65, 1995.

[Rector & al. 1994] A. L. Rector, W. D. Solomon, W. A. Nowlan, et T. W. Rush, A Terminology Server for Medical Language and Medical Information Systems, *Methods of Information in Medicine*, Vol. 34, pp. 147-157, 1994.

[Reidsma & al. 2003] D. Reidsma, J. Kuper, T. Declerck, H. Saggion et H. Cunningham, Cross Document Ontology based Information Extraction for Multimedia Retrieval, *Supplementary Proceedings of the 11th International Conference on Conceptual Structures (ICCS'03)*, Dresden, Germany, 2003.

[Rey 2003] A. Rey, Petit Robert de la langue française, Editions Le Robert, ISBN 285036826-1, 2003.

[Riloff 1993] E. Riloff., Automatically constructing a dictionary for information extraction tasks, *Proceedings of the Eleventh National Conference on Artificial Intelligence*, Whashington D.C., USA, pp. 811-816, 1993.

[Riloff 1996] E. Riloff, Automatically generating extraction patterns from untagged text, *Proceedings of the 13th International Conference on Artificial Intelligence (AAAI'96)*, Portland, USA, pp. 1044-1049, 1996.

[Rousselot & Frath 2002] F. Rousselot et P. Frath, Terminologie et Intelligence Artificielle, *Traits d'union (12èmes rencontres linguistiques en pays rhénans)*, G. Kleiber et N. Le Querler (réd.), Presses Universitaires de Caen, pp. 181-192, 2002.

[Sabah 2001] G. Sabah, Sens et traitements automatiques des langues (chapitre 3), *Ingénierie des langues*, J.-M. Pierrel (réd.), Hermès, p. 77-108, 2001.

[Sager 1990] J. Sager, A Pratical Course in Terminology Processing, John Benjamins Publishing Co., Amsterdam, Netherlands, 1990.

[Sager 1978] N. Sager, Natural Language Information Formatting: The Automatic Conversion of Texts to a Structured Data Base, *Advances in Computers*, M.C. Yovits, (réd.), Academic Press, New York, Vol. 17, pp. 89-162, 1978.

[Sager 1981] N. Sager, Natural Language Information Processing: A Computer Grammar of English and Its Applications, Addison-Wesley Publishing, Reading, Massachusetts, 1981.

[Sager & Lyman 1978] N. Sager et M. Lyman, Computerized Language Processing: Implications for Health Care Evaluation, *Medical Record News*, Vol. 49(3), pp. 20-30, 1978.

[Salton & al. 1975] G. Salton, A. Wong et C. S. Yang, A vector space model for automatic indexing, *Communications of the ACM*, Vol. 18(11), pp. 613–620, 1975.

[Salton & McGill 1983] G. Salton et M. J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill Book Company, 1983.

[Sartre 1943] J.-P. Sartre, *L'Être et le Néant, essai d'ontologie phénoménologique*, Gallimard, ISBN 207029388-2, 1943.

[Schank 1975] R. C. Shank (réd.), Conceptual Information Processing, Elsevier Science Publishers, 1975.

[Schneider & Renz 2000] R. Schneider et I. Renz, The relevance of frequency lists for error correction and robust lemmatization, *Actes des 5èmes journées internationales d'analyse statistique des données textuelles (JADT2000)*, Lausanne, Switzerland, pp. 43-50, 2000.

[Schreiber & al. 1993] G. Schreiber, B. Wielinga et J. Breuker (eds.), KADS - A Principled Approach to Knowledge-Based Systems Development, Academic Press, Londres, Grande-Bretagne, 1993.

[Scowen 1996] R. S. Scowen, Extended BNF – A generic base standard, ISO 14977, 1996.

[Shrihari & Li 2000] R. Srihari and W. Li, Information extraction supported question answering, *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*, Gaithesburg, Maryland, NIST Special Publication 500-246, pp. 185-196, 2000.

[Smadja 1993] F. A. Smadja, Retrieving collocations from text: Xtract, *Computational Linguistics*, Vol. 19(1), pp. 143-177, 1993.

[Soderland 1999] S. Soderland, Learning Information Extraction Rules for Semi-Structured and Free Text, *Machine Learning*, Vol. 34, pp. 233-272, 1999.

[Soderland & al. 1995a] S. G. Soderland, D. Fisher, J. Aseltine et W. Lehnert, CRYSTAL: Inducing a Conceptual Dictionary, *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95)*, Montréal, Canada, pp. 1314-1319, 1995.

[Soderland & al. 1995b] S. G. Soderland, D. Aronow, D. Fisher, J. Aseltine et W. Lehnert, Machine Learning of text analysis rules for clinical records, *Technical Report TE-39*, University of Massachusetts, 1995.

[Soubbotin 2001] M.M. Soubbotin, Patterns of Potential Answer expressions as Clues to the Right Answers, *Proceedings of the Tenth Text REtrieval Conference (TREC 2001)*, Gaithersburg, USA, NIST Special Publication 500-250, pp. 293-302, 2001.

[Sowa 1984] J. Sowa, Conceptual Structures: Information processing in Mind and Machine, Addison-Wesley Publishing, Reading, Massachusetts, 1984.

[Staab & al. 1999] S. Staab, C. Braun, A. Düsterhöft, A. Heuer, M. Klettke, S. Melzig, G. Neumann, B. Prager, J. Pretzel, H.-P. Schnurr, R. Studer, H. Uszkoreit et B. Wrenger. GETESS -- Searching the Web Exploiting German Texts, *Proceedings of the 3rd Workshop on Cooperative Information Agents (CIA '99)*, Upsala, Sweden, Lecture Notes in Computer Science, LNCS 1652, Springer, pp. 113-124, 1999.

[Staab & al. 2001] S. Staab, H.-P. Schnurr, R. Studer et Y. Sure, Knowledge processes and ontologies, *IEEE Intelligent Systems*, *Special Issue on Knowledge Management*, Vol. 16(1), pp. 26-34, 2001.

[Sundheim 1991] B. M. Sundheim (réd.), MUC-3, *Proceedings of the Third Message Understanding Conference*, San Diego, USA, Morgan Kaufmann Publisher, 1991.

[Sundheim 1992a] B. M. Sundheim (réd.), MUC-4, *Proceedings of the Fourth Message Understanding Conference*, McLean, USA, Morgan Kaufmann Publisher, 1992.

[Sundheim 1992b] B. M. Sundheim, Overview of the fourth message understanding evaluation and conference, *Proceedings of the Fourth Message Understanding Conference*, McLean, USA, Morgan Kaufmann Publisher, 1992.

[Sundheim 1993a] B. M. Sundheim (réd.), MUC-5, *Proceedings of the Fifth Message Understanding Conference*, Baltimore, USA, Morgan Kaufmann Publisher, 1993.

[Sundheim 1993b] B. M. Sundheim, TIPSTER/MUC-5 information extraction system evaluation, *Proceedings of the Fifth Message Understanding Conference*, Baltimore, USA, Morgan Kaufmann Publisher, 1993.

[Sundheim 1995] B. M. Sundheim (réd.), MUC-6, *Proceedings of the Sixth Message Understanding Conference*, Columbia, USA, Morgan Kaufmann Publisher, 1995.

[Sure 2003] Y. Sure, Methodology, tools and case studies for ontology based knowledge management, *PhD Thesis*, Université de Karlsruhe, Germany, 2003.

[Swartout & al. 1997] B. Swartout, R. Patil, K. Knight et T. Russ, Towards distributed use of large-scale ontologies, *Ontological Engineering*, AAAI-97 Spring Symposium Series, Stanford, USA, pp. 138-148, 1997.

[Takeda & al. 2001] K. Takeda, H. Nomiyama, T. Nasukawa, M. Kobayashi, T. Sakairi, H. Matsuzawa, T. Nagano, A. Murakami et H. Takeuchi, Text Mining and Site Outlining Projects, *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium (NLPRS2001)*, Tokyo, Japon, pp. 773-774, 2001.

[Tartier 2000] A. Tartier, Evolution terminologique : méthodes d'analyse automatique de corpus diachroniques, *Actes de RECITAL 2000 (Rencontre des Etudiants Chercheurs en Informatique pour le Traitement Automatique des Langues Naturelles)*, Lausanne, Suisse, pp. 523-527, 2000.

[TIPSTER 1993] TIPSTER Program Phase I, *Proceedings of Tipster Text Program (Phase I)*, Morgan Kaufmann Publishers, San Francisco, USA, 1993.

[TIPSTER 1996] TIPSTER Program Phase II, *Proceedings of Tipster Text Program (Phase II)*, Morgan Kaufmann Publishers, San Francisco, USA, 1996.

[TIPSTER 1998] TIPSTER Program Phase III, *Proceedings of the Tipster Text Phase III Workshop*, Morgan Kaufmann Publishers, San Francisco, USA, 1998.

[Todirascu & al. 2002] A. Todirascu, L. Romary et D. Bekhouche, Vulcain - An Ontology-Based Information Extraction System, *Natural Language Processing and Information Systems, 7th International Conference on Applications of Natural Language to Information Systems (NLDB 2002),* B. Andersson, M. Bergholtz et P. Johannesson (réd.), Stockholm, Sweden, Lecture Notes in Computer Science, LNCS 2553, Springer, pp. 64-75, 2002.

[Tremblay 1998] L. Tremblay, La qualité de la langue et les médias écrits, *Norme et médias, Terminogramme 97-98*, D. Raymond et A. A. Lafrance (réd.), Office québécois de la langue française, 1998.

[TREC 1999] TREC-8, *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*, Gaithersburg, USA, NIST Special Publication 500-246, E. M. Voorhees et D. K. Harman (réd.), 1999.

[Trichet 1998] F. Trichet, DSTM : un environnement de modélisation et d'opérationalisation de la démarche de résolution de problèmes d'un Système à Base de Connaissances, *Thèse de Doctorat (Spécialité Informatique)*, Université de Nantes, 1998.

[Uschold 1996] M. Uschold, Building ontologies: towards a unified methodology, *Proceedings of the 16th Annual Conference of the British Computer Society Specialist Group on Expert Systems (Experts Systems 96)*, Cambridge, Great-Britain, 1996.

[Uschold & al. 1998] M. Uschold, M. King, S. Moralee et Y. Zorgios, The Enterprise Ontology. The Knowledge Engineering Review, *Special Issue on Putting Ontologies to Use*, M. Uschold et A. Tate (réd.), Vol. 13, 1998.

[Valarakos & al. 2003] A. Valarakos, G. Sigletos, V. Karkaletsis et G. Paliouras. A Methodology for Semantically Annotating a Corpus Using a Domain Ontology and Machine Learning, *Proceedings of the RANLP'2003 Conference*, Borovets, Bulgaria, pp. 494-499, 2003.

[Van Rijsbergen 1979] C. J. Van Rijsbergen, Information Retrieval, Butterworths, Londres, ISBN 040870929-4, 1979.

[Vergne 2003] J. Vergne, Un outil d'extraction terminologique endogène et multilingue, *Actes de TALN 2003 (10ème conférence sur le Traitement Automatique des Langues Naturelles)*, Tome 2, pp. 139-148, 2003.

[Véronis & Guimier De Neef 2004] J. Véronis et E. Guimier De Neef, Le traitement automatique des nouvelles formes de communication écrite (e-mails, forums, chats, SMS, etc.), *Journées d'Etude de l'Association pour le Traitement automatique des Langues (ATALA)*, `http://www.up.univ-mrs.fr/~veronis/je-nfce/`, ENST, Paris, 2004.

[Wakao & al. 1996] T. Wakao, R. Gaizauskas et Y. Wilks, Evaluation of an Algorithm for the Recognition and Classification of Proper Names, *Proceedings of the 16th International Conference on Computational Linguistics (COLING96)*, Copenhagen, Danmark, pp. 418-423, 1996.

[Wall & al. 2000] L. Wall, Y. Christiansen et J. Orwant, Programming Perl (Third Edition), ISBN 059600027-8, 2000.

[Wilks 1997] Y. Wilks, Information Extraction as a Core Language Technology, *Information Extraction (a multidisciplinary approach to an emerging technology)*, M. T. Pazienza (réd.), Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence subseries), LNAI 1299, Springer, pp. 1-9, 1997.

[Wilks & al. 1997] Y. Wilks, B. Slator et L. Guthrie, Electric words: dictionaries, computers and meaning, ACL-MIT Press series in Natural Language Processing, MIT Press, Cambridge, ISBN 026223182-4, 1997.

[Will 1993a] C. A. Will, Comparing Human and Machine Performance for Natural Language Information Extraction: Results from the TIPSTER Text Evaluation, *Proceedings of TIPSTER Text Program (Phase I)*, Morgan Kaufmann Publishers, San Francisco, pp. 179-194, 1993.

[Will 1993b] C. A. Will, Comparing Human and Machine Performance for Natural Language Information Extraction: Results for English Microelectronics from the MUC-5 Evaluation. *Proceedings of the Fifth Message Understanding Conference (MUC-5)*, Morgan Kaufman Publishers, pp. 53-67, 1993.

[Woods & Schmolze 1992] W. A. Woods et J. G. Schmolze, The KL-ONE family, *Computers and Mathematics with Applications*, Vol. 23 (2/5), pp. 133-177, 1992.

[Wüster 1976] E. Wüster, La théorie générale de la terminologie, un domaine interdisciplinaire impliquant la linguistique, la logique, l'ontologie, l'informatique et les sciences des objets, *Essai de définition de la terminologie*, *Actes du colloque international de terminologie*, H. Dupuis (réd.), Québec, Régie de la langue française, pp. 49-57, 1976.

[Wüster 1979] E. Wüster, Einführung in die allgemeine Terminologielehre und terminologische Lexicographie, New York, Springer, 1979.

[XTAG-Group 1995] The XTAG-Group, A Lexicalized Tree Adjoining Grammar for English, *Technical Report IRCS 95-03*, University of Pennsylvania, 1995.

[Yangarber & al. 2000] R. Yangarber, R. Grishman, P. Tapanainen et S. Huttunen, Automatic Acquisition of Domain Knowledge for Information Extraction, *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, Saarbrcken, Germany, pp. 940-946, 2000.

[Zipf 1965] George K. Zipf, Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology, New York: Hafner, 1965.

[Zweigenbaum & al. 1995] P. Zweigenbaum, B. Bachimont, J. Bouaud, J. Charlet et J.-F. Boisvieux, Issues in the structuration and acquisition of an ontology for medical language understanding, *Methods of Information in Medicine*, Vol. 34(1/2), pp. 15-24, 1995.

[Zweigenbaum & al. 2001] P. Zweigenbaum, P. Jacquemart, N. Grabar et B. Habert, Building a text corpus for representing the variety of medical language, *Proceedings of the tenth world congress on Medical Informatics (MedInfo 2001)*, V. L. Patel, R. Rogers et R. Haux (réd.), London, Great-Britain, 2001.

Annexes

Annexe A

Les cinq tâches MUC

Lors des sixièmes et septièmes conférences MUC (*Message Understanding Conference*) [Sundheim 1995] [MUC-7 1998], la tâche d'Extraction d'Information [Chinchor & Marsh 1998] a été découpée en cinq sous-tâches :

- la reconnaissance des entités nommées (*Named Entity* ou *NE*) ;
- la résolution de coréférences (*Coreference* ou *CO*) ;
- le remplissage de patrons d'entité (*Template Element* ou *TE*) ;
- la détection de relation (*Relation Template* ou *RT*) ;
- la description d'événement (*Scenario* Template ou *ST*).

Ce découpage est repris dans la définition de nombreux systèmes d'Extraction d'Information. Les quatre premiers types (*NE*, *TE*, *TR* et *ST*) sont les plus pertinents et les plus efficaces dans les systèmes d'Extraction d'Information. Le cinquième (*CO*) est utilisé en complément des quatre types précédents, son utilisation en dehors de ce cadre donnant des résultats très limités et peu significatifs.

A.1 Reconnaissance des entités nommées

La reconnaissance des entités nommées [Chinchor 1997] consiste à reconnaître dans les textes les noms de personnes, de lieux, d'organisations (`ENAMEX` ou *entity name expression*), les expressions temporelles comme les dates (`TIMEX`) ou les expressions numériques comme les valeurs monétaires ou les pourcentages (`NUMEX`). La reconnaissance des entités nommées est souvent dépendante de la nature du corpus, ainsi passer d'un ensemble de textes abordant un domaine particulier à un autre traitant d'un sujet totalement différent (par exemple passer de chroniques sportives à des articles financiers) nécessite en règle générale des adaptations des systèmes.

A.2 Résolution de coréférence

La résolution de coréférence [Hirschman & Chinchor 1997] permet d'améliorer l'identification des relations entre entités dans les textes, ces entités étant à la fois celles issues de la reconnaissance d'entités nommées et celles provenant de références anaphoriques aux premières.

Exemple A.1 (Résolution de coréférence)

« Isabelle se tenait à la fenêtre, elle regardait les passants... »

La coréférence lie le nom "Isabelle" au pronom "elle".

Cette méthode est très dépendante du domaine. En effet les relations entre les entités sont soit différentes soit décrites différemment selon le type de texte dans lesquelles elles apparaissent. De ce fait le système doit être conditionné pour un type de documents particulier, il est fortement inefficace s'il est utilisé avec un autre type.

A.3 Remplissage de patrons d'entité

Il s'agit d'une méthode basée sur la reconnaissance d'entités nommées et la résolution de coréférence qui permet de trouver des caractéristiques d'un objet. Elle consiste à rajouter aux entités nommées identifiées des informations descriptives extraites grâce aux liens trouvés par la résolution de coréférence. La nature des informations descriptives est donnée par les champs d'un *patron* (aussi appelé *formulaire*) dit d'entité (*template element*). Par exemple, pour un produit manufacturé, il s'agira de son nom, de la société qui le fabrique, de son prix, etc.

Exemple A.2 (Remplissage d'un patron d'entité)

```
Isabelle entendit frapper à la porte. Il était presque 21h et le soleil déclinait
doucement en ce doux soir d'août. Elle portait une robe bleue et était prête à
sortir...
```

Le remplissage d'un patron d'entité PERSONNE (défini avec les arguments « `nom` », « `sexe` », « `habillement` » et « `état` ») à partir du texte précédent, donne le résultat suivant :

```
<PERSONNE 1>   [
                    nom : Isabelle
                    sexe : féminin
                    habillement : robe bleue
                    état : prête à sortir
               ]
```

Ce patron est rempli par l'identification dans le texte une personne nommée "Isabelle", de sexe féminin ("elle" étant un pronom féminin), habillée d'une robe bleue et dont l'état est "prête à sortir".

Le formalisme des patrons dépend de la nature des informations à extraire ainsi que de la finalité de cette extraction. Par exemple le remplissage d'une base de données nécessite un haut degré de formalisme alors que la rédaction automatique d'un rapport de recherche d'informations demande à ce que celles-ci soient facilement lisibles par un lecteur humain et donc relativement peu formelles. Cette remarque concerne les patrons de relation et de scénario (cf. section A.4 et A.5).

A.4 Détection de relations

Cette tâche a pour but de trouver les relations pouvant exister entre les différents patrons d'entités identifiés par la tâche précédente. Il peut s'agir par exemple de la relation d'appartenance d'une classe particulière d'éléments à une autre (papillon et insectes), de la relation familiale entre deux individus (frère et sœur) ou encore de la relation de dirigeant entre un individu et un groupe de personnes (chef militaire, chef de service dans une entreprise). Les relations sont rangées dans des patrons dits de relation (*relation template*).

A.5 Description d'événements

Cette méthode lie les différents résultats du remplissage de patrons d'entité afin de produire les relations possibles entre ces patrons (comme dans la détection de relations) mais également les événements concernant certains de ces patrons et induits par des informations contenues dans les textes. On obtient ainsi une description des événements contenus dans les documents analysés. Comme pour les patrons d'entités, des patrons dits de scénario (*scenario template*) définissent les types de relation et d'entités formant les événements à identifier.

Exemple A.3 (Description d'événements)

```
Lors de son séjour à La Plagne la semaine dernière, Caroline, la soeur d'Isabelle
la jeune fleuriste de la rue Manfreton, s'est cassé la jambe à cause d'un accident
de ski et a été admise à l'hôpital. Les médecins lui ont annoncé qu'elle sortirait
d'ici deux semaines. Ainsi son mariage avec Paul prévu à la fin du mois prochain à
Lyon pourra avoir lieu normalement.
```

À partir du texte précédent, la phase de description d'événements peut permettre de remplir les patrons PERSONNE, RELATION et EVENEMENT décrits ci-dessous.

```
<PERSONNE>
[
  nom :
  état :
  profession :
]
```

```
<RELATION>
[
  nature :
  personnes concernées :
]
```

```
<EVENEMENT>
[
 nature :
 personnes concernées :
 état temporel :
]
```

L'exécution de la phase de description d'événement sur le texte donne les résultats suivants :

```
<PERSONNE 1>
[
  nom : Caroline
  état : jambe cassée
  profession :
]

<PERSONNE 2>
[
  nom : Isabelle
  état :
  profession : fleuriste
]

<PERSONNE 3>
[
  nom : Paul
  état :
  profession :
]

<RELATION 1>
[
  nature : fraternelle
  personnes concernées : <PERSONNE 1> <PERSONNE 2>
]

<EVENEMENT 1>
[
  nature : accident
  personnes concernées : <PERSONNE 1>
  état temporel : passé
]

<EVENEMENT 2>
[
  nature : mariage
  personnes concernées : <PERSONNE 1> <PERSONNE 3>
  état temporel : futur
]
```

Annexe B

L'étiqueteur de Brill

Un étiqueteur (ou catégoriseur) est un outil d'étiquetage automatique de textes qui affecte à chaque mot (ou à chaque terme) d'un texte une étiquette représentant la catégorie grammaticale lui correspondant dans le texte étudié.

B.1 Présentation

L'étiqueteur de Brill [1] [Brill 1993] de l'Université de Pennsylvanie aux États-Unis et est fondé sur le principe de l'apprentissage. Il s'agit de donner au départ à l'étiqueteur un échantillon du corpus à traiter, échantillon préalablement annoté par un expert humain. À partir de cet extrait, l'étiqueteur déduit un certain nombre de règles lexicales destinées à l'étiquetage des mots inconnus, et de règles contextuelles permettant de corriger les erreurs (mauvaises affectations d'étiquettes), règles utilisées pour effectuer l'étiquetage à proprement parler. L'étiqueteur a également recours à un lexique contenant un certain nombre de mots (ou termes) du corpus et les étiquettes leur correspondant. À partir de ce lexique et des règles lexicales, il effectue un premier étiquetage. Ensuite il revient sur ces premières affectations et les corrige, si besoin est, en examinant la situation locale de chaque mot et en appliquant les règles contextuelles afin qu'après ces traitements, chaque mot ait reçu l'étiquette correspondant à sa catégorie grammaticale dans son contexte.

L'étiqueteur développé par Eric Brill[2] a été d'abord testé sur l'anglais mais de part son principe de fonctionnement fondé sur l'apprentissage, il s'applique également à d'autres langues. Pour le français la version la plus communément utilisée est celle entraînée par l'INaLF[3] (*Institut National de la Langue Française*) [Lecomte & Paroubek 1996].

[1] `http://www.cs.jhu.edu/~brill`

[2] Eric Brill, Etiqueteur de Brill Version 1.14, Août 1994, disponible à cette adresse :
`ftp://ftp.cs.jhu.edu/pub/brill/Programs/RULE_BASED_TAGGER_V.1.14.tar.Z`

[3] `http://www.inalf.cnrs.fr`

B.2 Pré-apprentissage

Cette procédure est nécessaire lorsque l'on désire étiqueter un corpus particulier, c'est-à-dire traitant un sujet spécifique et de ce fait utilisant des mots propres à celui-ci. Ces mots ont de fortes chances d'être mal étiquetés en utilisant un étiqueteur entraîné sur un autre corpus.

Tout d'abord il faut définir un jeu d'étiquettes si l'on ne veut pas utiliser celles proposées par Brill. Ensuite on étiquette manuellement une (petite) partie du corpus qui aura été préalablement mise sous une forme acceptable pour Brill (ponctuations entourées d'espaces et une phrase par ligne), on appellera cet échantillon `TAGGED-CORPUS`. On crée ensuite les fichiers suivant :

- `UNTAGGED-CORPUS` : l'échantillon non étiqueté ;
- `TGD1`, `TGD2` : la première et la seconde moitié de l'échantillon étiqueté ;
- `BIGWORDLIST` : liste de tous les mots de `UNTAGGED-CORPUS` ;
- `BIGRAMLIST` : liste de toutes les paires de mots de `UNTAGGED-CORPUS` ;
- `SMALLWORDTAGLIST` : la liste des associations [mot code fréquence], donnant le nombre d'apparition d'un mot dans `TGD1` avec le même code.

Après la création on peut lancer la commande d'apprentissage des règles lexicales :

```
unknown-lexical-learn.prl BIGWORLDLIST SMALLWORDTAGLIST BIGRAMLIST 300
LEXRULEFILE
```

`LEXRULEFILE` est le fichier dans lequel figurent les règles lexicales nouvellement créées. Le nombre 300 est une valeur destinée à améliorer l'efficacité, indiquant au système de n'utiliser le contexte fourni par les bigrammes que lorsque l'un des deux mots fait partie des 300 mots les plus fréquents.

Après l'obtention des règles lexicales, on s'intéresse aux règles contextuelles. Celles-ci sont créées comme suit :

- ***Création des fichiers suivants :***
  - `TRAINING.LEXICON` : lexique d'entraînement issu de tous les textes étiquetés sauf `TGD2` ;
  - `FINAL.LEXICON` : lexique final issu de `TAGGED-CORPUS` (celui qui sera utilisé par l'étiqueteur pour étiqueter des textes) ;
  - `UNTGD2` : seconde moitié de `TAGGED-CORPUS` dont sont ôtées les étiquettes ;
  - `DUMMY-TGD2` : fichier résultant de l'étiquetage de `UNTGD2` par le système en se servant du lexique et des règles lexicales précédemment apprises. La comparaison de ce fichier avec `TGD2` va permettre au système d'extraire les règles contextuelles.

- ***Lancement de la commande des règles contextuelles :***

```
contextual-rule-learn TGD2 DUMMY-TGD2 CTXRULEFILE TRAINING.LEXICON
```

`CTXRULEFILE` est le fichier dans lequel figurent les règles contextuelles nouvellement apprises.

Ces commandes visant à permettre au système de créer sa propre base de connaissance sont décrites de façon plus complète dans les fichiers `README` d'Eric Brill, fichiers disponibles avec l'étiqueteur. Le fichier `README.TRAINING` en particulier détaille les commandes à effectuer et les petits outils (disponibles avec l'étiqueteur) à utiliser pour créer certains des fichiers précédemment cités.

Il est possible à l'utilisateur de modifier les règles précédentes où d'en ajouter d'autres *« à la main »*. Néanmoins cette opération doit être réalisée avec une extrême prudence, en effet il est très difficile d'évaluer l'impact réel d'une nouvelle règle sur l'efficacité de l'étiqueteur. Une règle apparemment utile peut se révéler être néfaste pour l'étiquetage (par exemple parce qu'elle rentre en contradiction avec d'autres règles). De plus, la position à laquelle on insère une règle est très importante vu que l'étiqueteur applique toutes les règles une par une dans l'ordre dans lequel elles se trouvent au sein du fichier qui les contient. *A contrario*, il est possible de modifier le lexique sans grand risque de générer de multiples erreurs. Il en va de même pour la liste de bigrammes (paires de mots adjacents).

B.3 Étiquetage d'un corpus

B.3.1 Prétraitements

Afin de pouvoir traiter un texte avec l'étiqueteur de Brill, celui-ci doit se présenter sous un certain formalisme. Il faut donc effectuer une mise en forme du texte afin de le conformer aux normes établies par Eric Brill.

Le texte doit être découpé de façon à obtenir une phrase par ligne, il faut enlever les balises textuelles ou tout autre attribut de mise en forme (gras, italique, etc.) et entourer chaque signe de ponctuation d'un espace. Il est à noter que l'étiqueteur de Brill est sensible à la casse.

D'autres prétraitements facultatifs peuvent également être réalisés selon la nature du corpus ou les desiderata de l'utilisateur (comme par exemple dans le cas de l'étiqueteur utilisé sur le français à l'InaLF, où les mots composés ou certaines associations de mots récurrentes sont traitées en reliant les différents constituants de ces expressions par un tiret-bas, formant ainsi des mots ajoutés dans le lexique afin qu'ils soient étiquetés correctement).

B.3.2 Processus d'étiquetage

La commande qui exécute le processus d'étiquetage est la suivante :

```
tagger LEXICON  LE_CORPUS  BIGRAMS LEXICALRULEFILE CTXRULEFILE
```

Cette commande fait intervenir les 4 fichiers de paramètres de l'étiqueteur plus celui contenant le corpus à étiqueter :

- `LEXICON` : une liste de mots dans laquelle chacun des mots est associé à une liste d'étiquettes, la première étant la plus probable, les autres (positionnées sans ordre particulier) sont celles qui ont été rencontrées pour ce mot dans le corpus (des étiquettes possibles). Ce fichier est le même que le fichier `FINAL.LEXICON` du paragraphe B.2 ;
- `LEXRULEFILE` : la liste des règles lexicales apprises par le système ;
- `CTXRULEFILE` : la liste des règles contextuelles apprises par le système, règles contenant les modèles de transformations contextuels qui vont servir à affiner l'étiquetage en contexte ;
- `BIGRAMLIST` : contient les paires de mots (bigrammes) repérées dans le corpus à étiqueter ;
- `LE_CORPUS` : contient le corpus à étiqueter.

L'étiquetage se déroule en trois étapes :

1. L'étiqueteur affecte à chaque mot son étiquette la plus probable en comparant le texte avec le lexique. Il existe une étiquette par défaut pour les mots inconnus (`NNP` pour les mots commençant par une majuscule, `NN` pour les autres).
2. Utilisation des règles lexicales : pour chaque mot inconnu (resté `NN` ou `NNP`) toutes les règles lexicales sont successivement essayées et appliquées si les conditions sont remplies (de ce fait chaque règle prend en compte le résultat précédemment acquis). Après cette étape, il peut rester certains mots étiquetés avec `NNP` ou `NN`, le système pouvant ne pas avoir trouvé de règles s'appliquant à ces mots.
3. Utilisation des règles contextuelles : le système applique systématiquement, sur le texte précédemment étiqueté, les modèles de transformations contextuels.

Le résultat se présente sous la forme d'une suite de phrases (une par ligne). Chaque mot est suivi d'une étiquette (de la forme `/PREP`). L'étiqueteur par défaut affiche le résultat à l'écran. Pour obtenir le texte étiqueté dans un fichier, il faut rediriger la commande d'étiquetage vers un fichier résultat.

Annexe C

Base de règles de SyGET pour le corpus [CREC]

Cette annexe présente un exemple d'une base de règles de SyGET c'est-à-dire des fichiers produits par le module de génération de la base de règles. Les fichiers ci-dessous ont été obtenus à partir des règles de grammaire formalisant l'ontologie d'extraction définie lors de l'expérimentation sur le corpus [CREC]. Dans cette base, le paramètre de séparation (cf. section 7.2.2.3) n'a été valué pour aucune des règles de type RC et RD.

C.1 Fichier « règles terminales »

```
' (banque nationale paris|banque populaire|bnp|bp|c epargne|c agricole|ca|caisse
epargne|ccf|cl|cn|cred maritime|cred mari|credit lyonnais|credit commercial de france|csse
epargne|credit agricole|caisse d'epargne|credit du nord|credit maritime|credit nord|sg|societe
generale) '
-> ' <C_BANQUE>$1</C_BANQUE> '

' (axa|azur|agf|crama|drouot|groupama|maaf|macif|maif|msa|mutuelles du mans|mutuelles
mans|mutuelles poitiers|mutuelles de poitiers|uap) '
-> ' <C_COMP_ASSURANCE>$1</C_COMP_ASSURANCE> '

' (cetelem|cnp|cofidis|finaref) '
-> ' <C_COMP_CREDIT>$1</C_COMP_CREDIT> '

' (alcatel|auchan) '
-> ' <C_DIV_COMP>$1</C_DIV_COMP> '

' (cm|cmlaco|cmut|cmb|cred mut|credit mutuel|je|la fede|ma part|notre part|notre|nous|on) '
-> ' <C_CREDIT_MUTUEL>$1</C_CREDIT_MUTUEL> '

' (moteur|abbateuse|cuisine amenage|cuisine amenagee|ordinateur|chauffage|materiel
informatique|informatique|micro-ordinateur|micro ordinateur|bateau) '
-> ' <C_DIV_PRODUIT>$1</C_DIV_PRODUIT> '

' (achat|rachat|acquisition|acheter|acquerir) '
-> ' <C_DC_ACHAT>$1</C_DC_ACHAT> '
```

```
' (vente|vendre) '
-> ' <C_DC_VENTE>$1</C_DC_VENTE> '

' (simulation|simuler) '
-> ' <DC_SIMULATION>$1</C_C_DC_SIMULATION> '

' (projet|projeter) '
-> ' <DC_PROJET>$1</C_C_DC_PROJET> '

' (refus|refuse|pas de suite|sans suite|pas suite) '
-> ' <DC_REFUS>$1</C_C_DC_REFUS> '

' (de sa part|de leur part|sa part|leur part) '
-> ' <CLIENT>$1</C_C_CLIENT> '

' (mme|mlle|madame|mademoiselle|monsieur) '
-> ' <IND_PERS>$1</C_C_IND_PERS> '

' (travaux) '
-> ' <DC_TVX>$1</C_C_DC_TVX> '

' (ravalement|amenagement|renovation|agrandissement|reparation|peinture) '
-> ' <TYPE_TVX>$1</C_C_TYPE_TVX> '

' (radiateur|mur|veranda|toiture) '
-> ' <OBJET_TVX>$1</C_C_OBJET_TVX> '

' (rue|avenue|place) ' -> ' <DC_ADRESSE>$1</C_C_DC_ADRESSE> '
 (nantes|orvault|prefailles|saint herblain|st herblain|paris) '
-> ' <NOM_VILLE>$1</C_C_NOM_VILLE> '

' (region|departement|dom|tom|canton) '
-> ' <DC_REGION>$1</C_C_DC_REGION> '

' (france|canada|islande|japon|usa|tats-unis|grande
bretagne|angleterre|suisse|groenland|amerique) '
-> ' <PAYS>$1</C_PAYS> '

' (loire-atlantique|bretagne|la reunion) '
-> ' <C_NOM_REGION>$1</C_NOM_REGION> '

' (europe|amerique sud|amerique nord|asie|asie du sud est|afrique|afrique nord|europe de
l'ouest|europe ouest|europe de l'est|europe est|artique|
antartique) '
-> ' <C_CONTINENT>$1</C_CONTINENT> '

' (immo|immobilier|habitat|habitation|imb|construction|ancien|neuf) '
-> ' <C_DC_IMMOBILIER>$1</C_DC_IMMOBILIER> '

' (maison|terrain) '
-> ' <C_NATURE_IMB>$1</C_NATURE_IMB> '

' (studio|appartement|appart|f1|f2|f3|t1|t2|t3) '
-> ' <C_APPARTEMENT>$1</C_APPARTEMENT> '

' (mobil-home|mobilhome|mobil home) '
-> ' <C_DIV_IMB>$1</C_DIV_IMB> '

' (residence principale|residence locative|residence secondaire|residence|logement
principal|logement locatif|logement secondaire|logement) '
-> ' <C_TYPE_IMB>$1</C_TYPE_IMB> '

' (epargne|epargner) '
-> ' <C_DC_EPARGNE>$1</C_DC_EPARGNE> '
```

```
' (compte cheque|cc|cc sans decouvert|cc avec decouvert standard|cc avec decouvert
personnalise|cc divers|cc Dailly|credit promoteur|compte courant|compte cheque sans
decouvert|compte cheque avec decouvert standard|compte cheque avec decouvert
personnalise|compte cheque divers|compte cheque Dailly) '
-> ' <C_COMPTE_CHEQUE>$1</C_COMPTE_CHEQUE> '

' (compte a terme|compte terme|compte a terme revenu|compte terme revenu|compte a terme
liberte|compte terme liberte|compte a terme expansion|
compte terme expansion) '
-> ' <C_COMPTE_A_TERME>$1</C_COMPTE_A_TERME> '

' (cel|lee|pel) '
-> ' <C_EPARGNE_LOGEMENT>$1</C_EPARGNE_LOGEMENT> '

' (pep|pep garanti|pep liberte|pep revenu|pep projet|pep livret retraite|
pep rubis) '
-> ' <C_PEP>$1</C_PEP> '

' (livret|livret bleu|livret jeune|livret epargne populaire|2eme livret|deuxieme livret|2e
livret|codevi) '
-> ' <C_LIVRET>$1</C_LIVRET> '

' (bon de caisse|bon caisse|bon epargne|bon d'epargne) '
-> ' <C_BON_BANCAIRE>$1</C_BON_BANCAIRE> '

' (capital revenu|capital expansion) '
-> ' <C_GAMME_CAPITAL>$1</C_GAMME_CAPITAL> '

' (per|per gestion libre) '
-> ' <C_PRODUIT_PER>$1</C_PRODUIT_PER> '

' (acm|orchidee|orchidee 4,5|orchidee 5|orchidee 6,3|orchidee 7,2|orchidee 8,3|orchidee
9,82|livret retraite|livret retraite 94|capucine|capucine 4,5|capucine 5|capucine 6,3|capucine
7,2|titane 2|surcapi|surepargne) '
-> ' <C_ACM>$1</C_ACM> '

' (suravenir|previ-action|previ action|previaction|previ-retraite|previ
retraite|previretraite|previ-capital|previ capital|previcapital|previ-croissance|previ
croissance|previcroissance|previ-option|previ option|previoption|previ-independant|previ
independant|previindependant) '
-> ' <C_SURAVENIR>$1</C_SURAVENIR> '

' (carte bleue|carte bleue visa|carte bleue visa premier|mastercard|eurocard|mastercard
gold|cb|cb visa|cb visa premier|visa|visa premier|carte visa|carte visa premier|carte
eurocard|carte mastercard|carte mastercard gold|carte maestro|carte cirrus|carte epargne|carte
retrait|carte de retrait|chequier|chequier franc|chequier euro) '
-> ' <C_MOYEN_PAIEMENT>$1</C_MOYEN_PAIEMENT> '

' (domibanque|compte actif|pea|compte titre|titre|compte liquidite pea|liquidite
pea|sicav|autorisation decouvert|autorisation de decouvert) '
-> ' <C_DIV_SERV_BANCASS>$1</C_DIV_SERV_BANCASS> '

' (pret|credit) '
-> ' <C_PRET>$1</C_PRET> '

' (pret personnel|pret perso) '
-> ' <C_P_PERSO>$1</C_P_PERSO> '

' (pret conso|pret consommation|ppc) '
-> ' <C_P_DIV_CONSO>$1</C_P_DIV_CONSO> '

' (pret ammenagement habitat|phab|modulimmo|pret ministere logement|
credit relais) '
-> ' <C_P_IMMO>$1</C_P_IMMO> '
```

```
' (pret etude|pret etudiant) '
-> ' <C_P_ETUDE>$1</C_P_ETUDE> '

' (credit relais prof|credit relais professionnel|tresor prof|tresor professionnel|pret
artisanat|pret artisan|credit financement stock prof|credit financement stock
professionnel|pret equipement prof|pret equipement professionnel|pbe|pret livret epargne
entreprise) '
-> ' <C_PRET_PROF>$1</C_PRET_PROF> '

' (pret bonifie agri|pret bonifie agriculture|pret equipement agri|pret equipement
agriculture|relais agri|relais agriculture|pret conventionne agri|pret conventionne
agriculture|credit financement stock agri|credit financement stock agriculture|pbe agri|pbe
agriculture) '
-> ' <C_PRET_AGRI>$1</C_PRET_AGRI> '

' (credit relais asso|credit equipement asso|pse associatif) '
-> ' <C_PRET_ASSO>$1</C_PRET_ASSO> '

' (collectivite publique eig|pret org hlm eig|eig etablissements financiers|credit
collectivite codevi|tresor collectivite publique|credeco) '
-> ' <C_PRET_DIV_ORG>$1</C_PRET_DIV_ORG> '

' (credimmedia|preference) '
-> ' <C_REVOLVING>$1</C_REVOLVING> '

' (credit bail|loa|location finaciere) '
-> ' <C_SODELEM>$1</C_SODELEM> '

' (assurance habitat personnalise|assurance habitat standard|assurance habitat
confort|assurance mrh|corail eco|corail 2000|corail 3000|corail|aurore|aurore 2000) '
-> ' <C_ASS_MRH>$1</C_ASS_MRH> '

' (groupe maladie|gr maladie conj cmo|lilas|lilas senior|lilas plus|iris|edelweiss|assur
hospi|assur-hospi|assurhospi|pcm|gpcm entreprise|allocataire cirps|rc sante|ij hospi) '
-> ' <C_ASS_SANTE>$1<C_/C_ASS_SANTE> '

' (assurance vie|assur-capital|assur capital|assurcapital|jonquille|assur
obseque|primevere|assurance decouvert compte cheque|rer|rente education rev|myosotis|myosotis
prevoyance|myosotis incapactite|myosotis deces|zenith|mimosa|mimosa 12|mimosa
14|gentiane|assurance emprunteur|assurance accident|assur accident|assur-accident|azur|rente
vermeil) '
-> ' <ASS_PREVOYANCE>$1<C_/C_ASS_PREVOYANCE> '

' (assurance flotte automobile|assurance marchandise transportee|assurance amt) '
-> ' <C_ASS_AUTO>$1</C_ASS_AUTO> '

' (assurcarte|assur carte|assur'carte|assurcarte|hermine|acajou|acajou commercant|acajou
plus|rc scolaire|rc association|aida|assurance bris machine|assurance bris de
machine|fedebail|rc chasse|assurance navigation plaisance|assurance multi
association|assurance scolaire|assurance chasse|rec organisateur|rc vie privee|protection
juridique) '
-> ' <DIV_ASS>$1</C_DIV_ASS> '

' (assurance) '
-> ' <C_DC_ASSURANCE>$1</C_DC_ASSURANCE> '

' (auto|automobile|voiture) '
-> ' <C_DC_AUTO>$1</C_DC_AUTO> '

' (break|monospace|hdi) '
-> ' <C_TYPE_AUTO>$1</C_TYPE_AUTO> '
```

```
' (ax|bx|cx|zx|106|206|306|406|205|406|607|clio|twingo|kangoo|safrane|megane
|fiesta|cordoba|evasion|xantia|mondeo|sierra|xsara|scenic|multipla|saxo|picasso|r
25|bravo|brava|clio 2|laguna|laguna 2|coccinelle) '
-> ' <C_MODELE_AUTO>$1</C_MODELE_AUTO> '

' (toyota|peugeot|simca|citroen|ford|renault|fiat|mercedes|volvo|mercedes|bmw|
volkswagen|audi) '
-> ' <C_MARQUE_AUTO>$1</C_MARQUE_AUTO> '

' (moto|cyclo|cyclo moteur|mobyl|mobylette|scooter) '
-> ' <C_MOTO>$1</C_MOTO> '

' (camion|caravane|camping car|camping-car|tracteur|velo|motoculteur|voiturette) '
-> ' <C_DIV_VEH>$1</C_DIV_VEH> '

' (vehicule) '
-> ' <C_DC_VEHICULE>$1</C_DC_VEHICULE> '
```

C.2 Fichier « règles constitutives concept »

```
RS1     AUTO|MOTO|C_DIV_VEH|C_DC_VEHICULE     ->      C_VEHICULE
RS2     C_DC_AUTO       ->      C_AUTO
RS3     C_DC_IMMOBILIER->       C_IMMOBILIER
RS4     C_APPARTEMENT|C_DIV_IMB|C_DC_NATURE_IMB       ->      C_NATURE_IMB
RS5     C_TYPE_COMPAGNIE|C_DC_COMPAGNIE       ->      C_COMPAGNIE
RS6     C_BANQUE|C_COMP_ASSURANCE|C_COMP_CREDIT|C_DIV_COMP|C_CREDIT_MUTUEL  ->
        C_TYPE_COMPAGNIE
RS7     C_ASSURANCE|C_EPARGNE|C_SERVICE|C_DIV_PROD_BANCASS   ->      C_PROD_BANCASS
RS8     C_DC_EPARGNE|C_EPARGNE_BANCAIRE|C_EPARGNE_ASSURANCE  ->      C_EPARGNE
RS9     C_COMPTE_GEN|C_EPARGNE_LOGEMENT|C_PEP|C_LIVRET|C_DIV_EP_BANC->      C_EPARGNE_BANCAIRE
RS10    C_COMPTE_CHEQUE|C_COMPTE_A_TERME      ->      C_COMPTE_GEN
RS11    C_DIV_EP|C_DC_PEL       ->      C_EPARGNE_LOGEMENT
RS12    C_BON_BANCAIRE|C_GAMME_CAPITAL|C_PRODUIT_PER ->      C_DIV_EP_BANC
RS13    C_ACM|C_SURAVENIR       ->      C_EPARGNE_ASSURANCE
RS14    C_MOYEN_PAIEMENT|C_DIV_SERV_BANCASS   ->      C_SERVICE_BANCASS
RS16    C_P_CONSO|C_P_IMMO|C_P_ETUDE|C_P_TVX|C_P_ORG|C_REVOLVING|C_SODELEM
        ->      C_PRET
RS17    C_P_AUTO|C_P_PERSO|C_P_DIV_CONSO      ->      C_P_CONSO
RS18    C_PRET_PROF|C_PRET_AGRI|C_PRET_ASSO|C_PRET_DIV_ORG   ->      C_P_ORG
RS19    C_ASS_AUTO|C_ASS_MRH|C_ASS_SANTE|C_ASS_PREVOYANCE|C_DIV_ASS|C_DC_ASSURANCE
        ->      C_ASSURANCE
RS20    C_ACTION_BANCASS|C_SIMULATION|C_ACTION_INFO|C_ACHAT|C_VENTE|C_CHANGEMENT|
        C_DIV_ACTION    ->      C_ACTION
RS21    C_IND_PERS      ->      C_PERSONNE
RS22    C_DESIGNATION_TVX       ->      C_TRAVAUX
RS23    C_DC_TVX|C_TYPE_TVX     ->      C_DESIGNATION_TVX
RS24    C_ADRESSE|C_VILLE|C_REGION|C_PAYS|C_CONTINENT->      C_LIEU
RS25    C_NOM_VILLE|C_CODE_POSTAL       ->      C_VILLE
RS26    C_DC_REGION|C_NOM_REGION        ->      C_REGION
RS27    C_TYPE_AUTO|C_MARQUE_AUTO|C_MODELE_AUTO       ->      C_AUTO
RS28    C_NATURE_IMB|C_TYPE_IMB       ->      C_IMMOBILIER

RD3     C_NOM|C_PRENOM ->       C_PERSONNE
RD1     C_TYPE_AUTO|+C_MARQUE_AUTO|C_+MODELE_AUTO     ->      C_AUTO
RD2     C_NATURE_IMB|+C_TYPE_IMB        ->      C_IMMOBILIER
RD3     C_NOM|+C_PRENOM->       C_PERSONNE

RC1     C_PRET+C_IMMOBILIER     ->      C_P_IMMO
RC2     C_PRET+C_EPARGNE_LOGEMENT       ->      C_P_IMMO
RC3     C_PRET+C_TRAVAUX        ->      C_P_TVX
RC4     C_PRET+C_VEHICULE       ->      C_P_AUTO
RC5     C_DC_ASSURANCE+C_VEHICULE       ->      C_ASS_AUTO
RC6     C_DC_ASSURANCE+C_IMMOBILIER     ->      C_ASS_MRH
```

```
RC7     C_DC_VEHICULE+C_VEHICULE      ->      C_VEHICULE
RC8     C_DC_EMETTEUR+C_PERSONNE      ->      C_EMETTEUR_PHYSIQUE
RC9     C_DC_EMETTEUR+C_BANQUE ->     C_EMETTEUR_BANQUE
```

C.2.1 Fichier « règles sélectives 1 »

```
RS1     C_AUTO|C_MOTO|C_DIV_VEH       ->      C_VEHICULE
RS2     C_DC_AUTO       ->      C_AUTO
RS3     C_DC_IMMOBILIER->       C_IMMOBILIER
RS4     C_APPARTEMENT|C_DIV_IMB|C_DC_NATURE_IMB       ->      C_NATURE_IMB
RS5     C_TYPE_COMPAGNIE|C_DC_COMPAGNIE       ->      C_COMPAGNIE
RS6     C_COMP_ASSURANCE|C_COMP_CREDIT|C_DIV_COMP     ->      C_TYPE_COMPAGNIE
RS7     C_ASSURANCE|C_EPARGNE|C_SERVICE|C_DIV_PROD_BANCASS    ->      C_PROD_BANCASS
RS8     C_DC_EPARGNE|C_EPARGNE_BANCAIRE|C_EPARGNE_ASSURANCE   ->      C_EPARGNE
RS9     C_COMPTE_GEN|C_PEP|C_LIVRET|C_DIV_EP_BANC     ->      C_EPARGNE_BANCAIRE
RS10    C_COMPTE_CHEQUE|C_COMPTE_A_TERME      ->      C_COMPTE_GEN
RS11    C_DIV_EP|C_DC_PEL       ->      C_EPARGNE_LOGEMENT
RS12    C_BON_BANCAIRE|C_GAMME_CAPITAL|C_PRODUIT_PER ->       C_DIV_EP_BANC
RS13    C_ACM|C_SURAVENIR       ->      C_EPARGNE_ASSURANCE
RS14    C_MOYEN_PAIEMENT|C_DIV_SERV_BANCASS    ->      C_SERVICE_BANCASS
RS15    C_PRET_AUTO|C_PRET_PERSO|C_PRET_DIV_CONSO     ->      C_PRET_CONSO
RS16    C_PRET_PROF|C_PRET_AGRI|C_PRET_ASSO|C_PRET_DIV_ORG    ->      C_PRET_ORG
RS17    C_ASS_AUTO|C_ASS_MRH|C_ASS_SANTE|C_ASS_PREVOYANCE|C_DIV_ASS ->       C_ASSURANCE
RS18    C_ACTION_BANCASS|C_SIMULATION|C_ACTION_INFO|C_ACHAT|C_VENTE|C_CHANGEMENT|
        C_DIV_ACTION    ->      C_ACTION
RS19    C_IND_PERS|C_NOM|C_PRENOM       ->      C_PERSONNE
RS20    C_DESIGNATION_TVX       ->      C_TRAVAUX
RS21    C_DC_TVX|C_TYPE_TVX     ->      C_DESIGNATION_TVX
RS22    C_ADRESSE|C_VILLE|C_REGION|C_PAYS|C_CONTINENT->       C_LIEU
RS23    C_NOM_VILLE|C_CODE_POSTAL       ->      C_VILLE
RS24    C_DC_REGION|C_NOM_REGION        ->      C_REGION
RS25    C_PRET_CONSO|C_PRET_IMMO|C_PRET_ETUDE|C_PRET_TVX|C_PRET_ORG|C_REVOLVING
        |C_SODELEM      ->      C_DF_PRET
```

C.2.2 Fichier « règles sélectives 2 »

```
RS1     C_DC_VEHICULE   ->      C_VEHICULE
RS2     C_TYPE_AUTO     ->      C_AUTO
RS3     C_MARQUE_AUTO   ->      C_AUTO
RS4     C_MODELE_AUTO   ->      C_AUTO
RS5     C_NATURE_IMB    ->      C_IMMOBILIER
RS6     C_TYPE_IMB      ->      C_IMMOBILIER
RS7     C_C_EPARGNE_LOGEMENT    ->      C_EPARGNE_BANCAIRE
RS8     C_DC_ASSURANCE ->       C_ASSURANCE
RS9     C_NOM   ->      C_PERSONNE
RS10    C_PRENOM        ->      C_PERSONNE
RS11    C_BANQUE        ->      C_TYPE_COMPAGNIE
RS12    C_CREDIT_MUTUEL->       C_TYPE_COMPAGNIE
```

C.2.3 Fichier « règles disjonctives »

```
RD1     C_TYPE_AUTO|+C_MARQUE_AUTO|+C_MODELE_AUTO     ->      C_AUTO
RD2     C_NATURE_IMB|+C_TYPE_IMB      ->      C_IMMOBILIER
RD3     C_NOM|+C_PRENOM->       C_PERSONNE
```

C.2.4 Fichier « règles conjonctives »

```
RC1     C_DC_PRET+C_IMMOBILIER ->      C_PRET_IMMO
RC2     C_DC_PRET+C_EPARGNE_LOGEMENT  ->      C_PRET_IMMO
RC3     C_DC_PRET+C_TRAVAUX    ->      C_PRET_TVX
RC4     C_DC_PRET+C_VEHICULE   ->      C_PRET_AUTO
RC5     C_DC_ASSURANCE+C_VEHICULE     ->      C_ASS_AUTO
RC6     C_DC_ASSURANCE+C_IMMOBILIER   ->      C_ASS_MRH
RC7     C_DC_VEHICULE+C_VEHICULE      ->      C_VEHICULE
RC8     C_DC_EMETTEUR+C_PERSONNE      ->      C_EMETTEUR_PHYSIQUE
RC9     C_DC_EMETTEUR+C_BANQUE ->      C_EMETTEUR_BANQUE
RC10    C_DC_EMETTEUR+C_CREDIT_MUTUEL ->      C_EMETTEUR_CREDIT_MUTUEL
```

C.3 Fichier « règles informatives »

```
C_PRET   :     C_DF_PRET
               C_PRET_duree : C_DUREE
               C_PRET_taux : C_TAUX
               C_PRET_montant : C_SOMME
```

C.4 Fichier « règles prédicatives »

```
C_PROJET  :    C_DC_PROJET
               C_PROJET_objet : C_ACHAT|C_VENTE|C_IMMOBILIER|C_VEHICULE
                                |C_ACTION_BANCAIRE
               C_PROJET_datation :: C_DATE
               C_PROJET_localisation : C_LIEU
               C_PROJET_montant :: C_SOMME

C_ACHAT   :    C_DC_ACHAT
               C_ACHAT_objet : C_IMMOBILIER|C_VEHICULE|C_PROD_BANCAIRE
               C_ACHAT_datation : C_DATE
               C_ACHAT_localisation : C_LIEU
               C_ACHAT_montant : C_SOMME

C_VENTE   :    C_DC_VENTE
               C_VENTE_objet : C_IMMOBILIER|C_VEHICULE|C_PROD_BANCASS
               C_VENTE_datation : C_DATE
               C_VENTE_localisation : C_LIEU
               C_VENTE_montant : C_SOMME

C_REFUS   :    C_DC_REFUS
               C_REFUS_objet : C_PROJET
               C_REFUS_datation : C_DATE
               C_REFUS_emetteur : C_EMETTEUR_PHYSIQUE|C_EMETTEUR_BANQUE|C_CLIENT
                                  |C_EMETTEUR_CREDIT_MUTUEL
```

Extraction d'Information et modélisation de connaissances à partir de Notes de Communication Orale

Fabrice EVEN

Résumé

Malgré l'essor de l'Extraction d'Information et le développement de nombreuses applications dédiées lors de ces vingt dernières années, cette tâche rencontre des problèmes lorsqu'elle est réalisée sur des textes atypiques comme des Notes de Communication Orale.
Les Notes de Communication Orale sont des textes issus de prises de notes réalisées lors d'une communication orale (entretien, réunion, exposé, etc.) et dont le but est de synthétiser le contenu informatif de la communication. Leurs contraintes de rédaction (rapidité et limitation de la quantité d'écrits) sont à l'origine de particularités linguistiques auxquelles sont mal adaptées les méthodes classiques de Traitement Automatique des Langues et d'Extraction d'Information. Aussi, bien qu'elles soient riches en informations, elles ne sont pas exploitées par les systèmes extrayant des informations à partir de textes.
Dans cette thèse, nous proposons une méthode d'extraction adaptée aux Notes de Communication Orale. Cette méthode, nommée MEGET, est fondée sur une ontologie modélisant les connaissances contenues dans les textes et intéressantes du point de vue des informations recherchées (« ontologie d'extraction »). Cette ontologie est construite en unifiant une « ontologie des besoins », décrivant les informations à extraire, avec une « ontologie des termes », conceptualisant les termes du corpus à traiter liés avec ces informations. L'ontologie des termes est élaborée à partir d'une terminologie extraite des textes et enrichie par des termes issus de documents spécialisés. L'ontologie d'extraction est représentée par un ensemble de règles formelles qui sont fournies comme base de connaissance au système d'extraction SYGET. Ce système procède d'abord à un étiquetage des instances des éléments de l'ontologie d'extraction présentes dans les textes, puis extrait les informations recherchées. Cette approche est validée sur plusieurs corpus.

Mots-clés : Extraction d'Information, Note de Communication Orale, Traitement Automatique des Langues Naturelles, Ontologie, Modélisation, Terminologie

Abstract

In spite of the rise of Information Extraction and the development of many applications in the last twenty years, this task encounters problems when it is carried out on atypical texts such as oral communication notes.
Oral communication notes are texts which are the result of an oral communication (meeting, talk, etc.) and they aim to synthesize the informative contents of the communication. These constraints of drafting (speed and limited amount of writing) lead to linguistic characteristics which the traditional methods of Natural Language Processing and Information Extraction are badly adapted to. Although they are rich in information, they are not exploited by systems which extract information from texts.
In this thesis, we propose an extraction method adapted to oral communication notes. This method, called MEGET, is based on an ontology which depends on the information to be extracted ("extraction ontology"). This ontology is obtained by the unification of an "ontology of needs", which describe the information to be found, with an "ontology of terms" which conceptualize the terms of the corpus which are related to the required information. The ontology of terms is elaborated from terminology extracted from texts and enriched by terms found in specialized documents. The extraction ontology is formalized by a set of rules which are provided as a knowledge base for the extraction system SYGET. This system (1) carries out a labelling of each instance of every element of the extraction ontology and (2) extracts the information. This approach is validated in several corpora.

Keywords: Information Extraction, Oral Communication Note, Natural Language Processing, Ontology, Modelling, Terminology