

Description et évaluation de lexiques  
prédicatifs pour la génération  
semi-automatique de grammaires d'extraction

Aurélien Bossard

6 septembre 2007



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Cadre applicatif . . . . .	1
1.2	Contexte de l'étude . . . . .	2
1.3	Problématique . . . . .	3
1.4	Plan du mémoire . . . . .	4
<b>2</b>	<b>Aperçu des Ressources</b>	<b>5</b>
2.1	Généralités sur les verbes . . . . .	5
2.1.1	Le prédicat verbal et les arguments . . . . .	5
2.1.2	Valence, Cadres de Sous-Catégorisation et Structure Argumentale . . . . .	6
2.1.3	Rôles thématiques et rôles sémantiques . . . . .	6
2.1.4	Les alternances . . . . .	8
2.1.5	Les classes de Levin . . . . .	8
2.2	Ressources pour l'anglais . . . . .	9
2.2.1	VerbNet . . . . .	9
	Les rôles thématiques dans VerbNet . . . . .	9
	La polysémie dans VerbNet . . . . .	9
	Les alternances dans VerbNet . . . . .	10
2.2.2	PropBank . . . . .	11
2.2.3	FrameNet . . . . .	13
	Les cadres au sens de FrameNet . . . . .	14
	Les rôles sémantiques dans FrameNet . . . . .	15
	Les alternances et arguments de verbe dans FrameNet . . . . .	15
	Conclusions sur FrameNet . . . . .	16
2.3	Ressources pour le Français . . . . .	17
2.3.1	Les Voisins de Le Monde . . . . .	18
2.3.2	Les Tables du LADL . . . . .	19
	Les réalisations syntaxiques des verbes dans les tables du LADL . . . . .	20
	Les rôles des arguments dans les tables du LADL . . . . .	20
	La logique de constitution des tables du LADL . . . . .	21
	La sémantique dans les tables du LADL . . . . .	21

	Conclusions sur les tables du LADL . . . . .	23
2.3.3	Volem . . . . .	23
	Description sémantique des entrées lexicales et polysémie dans Volem . . . . .	23
	Les variations syntaxiques dans Volem . . . . .	24
	Les rôles thématiques dans Volem . . . . .	25
	Conclusions sur Volem . . . . .	27
2.3.4	Conclusions générales sur les ressources françaises . . . . .	27
<b>3</b>	<b>Choix d'une ressource pour l'extraction</b>	<b>29</b>
3.1	Description de la méthode d'extraction . . . . .	29
3.2	Descr dét. pour l'analyse des besoins . . . . .	31
3.2.1	Sélection des verbes . . . . .	31
3.2.2	Quel type de schémas de sous-catégorisation ? . . . . .	33
3.2.3	Etiquetage sémantique . . . . .	34
3.3	Le choix de la ressource . . . . .	34
3.3.1	Vers une nouvelle ressource plus adaptée à l'extraction d'information ? . . . . .	34
3.3.2	Volem ou les Tables du LADL ? . . . . .	35
	La gestion des alternances dans Volem et les Tables du LADL . . . . .	35
	La gestion de la polysémie dans les deux ressources . . . . .	37
	Volem et les Tables du LADL face à la sémantique des verbes . . . . .	38
	Conclusions . . . . .	39
<b>4</b>	<b>Réalisations</b>	<b>43</b>
4.1	Enrichissement et codage de Volem . . . . .	43
4.1.1	Le codage de Volem sous une forme plus appropriée à la génération automatique de patrons d'extraction . . . . .	43
	Le codage des informations de volem . . . . .	45
	Le codage des enrichissements apportés à Volem pour l'extraction . . . . .	45
4.1.2	L'auxiliaire . . . . .	46
4.1.3	Des rôles thématiques aux rôles sémantiques . . . . .	48
4.1.4	Les prépositions . . . . .	49
4.2	Automates d'extraction . . . . .	51
4.2.1	Format de sortie des automates . . . . .	51
4.2.2	Automates générateurs . . . . .	52
	Méthode de création d'automates et exemples . . . . .	52
	Précisions sur la méthode de création des automates . . . . .	54
4.2.3	Automates de récupération des prépositions et programmes liés . . . . .	54
4.3	Filtrage probabiliste des alternances . . . . .	55

4.4	Ajout d'arguments aux informations à extraire . . . . .	56
4.5	L'annotation des arguments . . . . .	57
4.6	La constitution de corpus . . . . .	58
4.7	Conclusion des réalisations . . . . .	58
<b>5</b>	<b>Evaluations</b>	<b>63</b>
5.1	Extraction des prépositions . . . . .	63
5.2	Les auxiliaires . . . . .	64
5.3	L'extraction d'informations . . . . .	64
5.3.1	Protocole d'évaluation . . . . .	64
5.3.2	Résultats de l'évaluation selon le protocole établi . .	64
5.3.3	Les formes syntaxiques existantes en corpus et non reconnues . . . . .	66
5.3.4	Un filtrage des alternances inutiles est-il possible? . .	66
<b>6</b>	<b>Conclusions et perspectives</b>	<b>71</b>
6.1	Conclusions . . . . .	71
6.2	Perspectives . . . . .	72
<b>A</b>	<b>Automates créés</b>	<b>75</b>
<b>B</b>	<b>Stats sur les alternances</b>	<b>77</b>
<b>C</b>	<b>Stats sur les alternances</b>	<b>81</b>
<b>D</b>	<b>Stats sur les adjonctions</b>	<b>85</b>



# Table des figures

2.1	Les rôles thématiques les plus communs de VerbNet . . . . .	10
2.2	Annotation PropBank de « buy » et « sell » . . . . .	12
2.3	Les réalisations syntaxiques de buy.v dans FrameNet . . . . .	15
2.4	Les réalisations syntaxiques de sell.v dans FrameNet . . . . .	16
2.5	Le contexte des réalisations syntaxiques de buy.v dans FrameNet . . . . .	17
2.6	Liste des relations du lemme acheter dans VDLM . . . . .	18
2.7	Liste des arguments du prédicat acheter obj de VDLM . . . . .	19
2.8	Les 15 premiers voisins d'acheter obj extraits de VDLM . . . . .	20
2.9	Extrait de la table 36DT des tables du LADL . . . . .	21
2.10	Etrait de la description de la table 36DT des tables du LADL . . . . .	22
2.11	Les différentes alternances de Volem (tirées de la doc en ligne de Volem) . . . . .	25
2.12	Les alternances du verbe acheter dans Volem . . . . .	26
2.13	Les rôles thématiques dans Volem . . . . .	26
2.14	Grille thématique du verbe acheter dans Volem . . . . .	27
2.15	Grille thématique du verbe voler dans Volem . . . . .	27
3.1	Les étapes de la génération de patrons d'extraction . . . . .	31
3.2	Extrait de la table 36DT des tables du LADL (bis) . . . . .	36
3.3	Les alternances du verbe acheter dans Volem (bis) . . . . .	37
4.1	Exemple d'une entrée du lexique de Volem avec le verbe « Acheter » . . . . .	44
4.2	Extraits de la table de données . . . . .	47
4.3	L'automate de repérage des auxiliaires . . . . .	48
4.4	Algorithme pour faire correspondre les rôles thématiques et les rôles sémantiques . . . . .	50
4.5	Méthode de récupération des prépositions . . . . .	50
4.6	Exemple de sortie au format des automates . . . . .	52
4.7	Exemples du passage d'alternances de Volem à des formes de surface . . . . .	54
4.8	Méthode pour enrichir Volem avec les prépositions . . . . .	55

4.9	Algorithme pour l'ajout d'adjonctions aux arguments à extraire	57
4.10	Exemple d'un automate générateur . . . . .	60
4.11	Exemple d'un graphe généré . . . . .	61
4.12	Brève description des différents corpus utilisés pour l'extraction	62
5.1	Protocole d'évaluation de l'extraction d'informations . . . . .	65

# Liste des tableaux

2.1	Le verbe cut dans VerbNet (extraits) . . . . .	11
2.2	Les cadres dans PropBank . . . . .	13
3.1	Arguments extraits d'une phrase du corpus FUSACQ . . . . .	30
3.2	Voisins de « acheter suj », « racheter suj » et « revendre à » dans VDLM . . . . .	40
3.3	Sélection de verbes grâce à la fusion de mesures de similarité des voisins de prédicats de VDLM . . . . .	41
5.1	Tableau de résultats chiffrés de l'extraction . . . . .	65
5.2	Répartition des alternances selon les verbes dans les phrases extraites des corpus (FirstInvest) . . . . .	67
5.3	Répartition des alternances selon les verbes dans les phrases extraites des corpus (FUSACQ annoté) . . . . .	68
5.4	Répartition des alternances selon les verbes dans les phrases extraites des corpus (Corpus général) . . . . .	69



# Chapitre 1

## Introduction

Afin de pouvoir automatiser les tâches de traitement du langage, il est nécessaire de disposer de ressources correspondant aux besoins de la tâche. La question des ressources est donc essentielle dans la réflexion sur le TAL. Il est important, lorsque l'on s'engage dans une tâche de traitement automatique du langage, de cerner les différents besoins en ressources afin de déterminer quel type de ressource est le plus adapté à la tâche à réaliser. Des réflexions sur la possibilité d'avoir pour le français une ressource qui satisfasse les besoins de la majorité des applications de TAL, sur le formalisme qu'il serait préférable d'utiliser afin de créer cette ressource, sont engagées, comme en témoigne la journée ATALA du 13 Mai 2006 sur les ressources françaises et l'éventualité de la création d'un « FrameNet » français.

### 1.1 Cadre applicatif

Le stage présenté dans ce mémoire a eu lieu au sein du LIPN dans le cadre du projet Infomagic. Le projet Infomagic s'inscrit dans le cadre du pôle de compétitivité IMVN (Image, multimédia et vie numérique). Ce pôle est le troisième d'Ile de France. Il est porté par l'agence régionale du développement Ile de France. Il est consacré aux technologies de l'information et de la communication et implique de grands groupes comme TF1, Lagardère Groupe, France Télécom, Eclair, SFP, TSF, des PME, des laboratoires et des institutions : LIP6 Paris VI, l'INA, l'IRCAM, Télécom Paris, CNAM, ESIEE, ENSTA, ENS Louis-Lumière, Gobelins, Femis.

Infomagic vise à mettre en place, sur une période de trois ans, un labo-

ratoire industriel de sélection, de tests, d'intégration et de validation d'applications opérationnelles des meilleures technologies franciliennes dans le domaine de l'ingénierie des connaissances.

Ce laboratoire s'appuie sur une plate-forme commune qui doit couvrir les grands domaines de l'analyse d'information, quelles que soient les sources (données structurées, texte, images et sons) :

- la recherche et l'indexation,
- l'extraction de connaissances,
- et la fusion d'informations multimédias.

Elle inclura des applications pour les secteurs de la e-Education et de la gestion des patrimoines culturels numériques.

Les partenaires du projet sont répartis en quatre catégories :

- Industriels : THALES (coordinateur), EADS, XEROX
- PME's : BERTIN, EUROPLACE, FISTCNRS, INTUILAB, ODILE JACOB, PERTIMM, TEMIS, VECSYS
- Etablissements publics : CEA, CNRS, INA, ONERA
- Ecoles et Universités : GETENST-INT, PARIS VI (LIP6, LSTA), PARIS VIII (LC&U), PARIS IX Dauphine (CEREMADE), PARIS XIII (LIPN), PARIS-SUD Orsay (LIMSI), UML Marne la Vallée (IGM), CNRS/LACAN

## 1.2 Contexte de l'étude

Ce mémoire vise à prendre connaissance des différentes ressources lexicales portant sur les verbes (nous nous restreindrons aux ressources pour l'anglais et aux ressources pour le français), de les analyser, pour finalement les évaluer dans le cadre d'une tâche d'extraction d'informations.

L'approche que nous utiliserons sera axée sur les verbes, car ceux-ci sont généralement porteurs d'une relation mettant en jeu leurs arguments. L'étude de la syntaxe d'une phrase peut permettre une mise en relation des arguments d'un verbe avec les rôles que ceux-ci jouent dans la relation que leur verbe porte. L'extraction d'informations que nous allons effectuer portera sur les rachats d'entreprise. Il s'agira d'extraire d'un corpus, sous la forme d'une base de données structurée, des phrases du type :

1. CPI achète Fulmar.
2. CPI a racheté Fulmar à son PDG pour 50 millions d'euros.
3. Rachat de Fulmar par CPI.
4. Ayant racheté Fulmar hier, CPI a ...

et éventuellement des structures comme :

1. CPI pourrait racheter Fulmar.
2. CPI a indiqué avoir racheté Fulmar.

En revanche, il ne convient pas d'extraire des structures comme :

1. Laurent s'est acheté un ordinateur. (pas un rachat d'entreprise)
2. CPI a acquis une grosse notoriété auprès... (sens métaphorique)

Il conviendra donc d'étudier la façon dont sont gérées les alternances, la polysémie, les propriétés des verbes en général, afin d'être en mesure de filtrer efficacement les entités à extraire.

### 1.3 Problématique

Une observation rapide permet de constater qu'il existe plus de ressources lexicales sur les verbes pour l'anglais que pour le français. Nous devons analyser les ressources pour l'anglais et pour le français afin d'être en mesure de répondre aux questions suivantes :

- Comment identifier en corpus les différents sens d'un verbe ?
- Les ressources actuelles pour l'anglais sont-elles suffisantes pour une tâche d'extraction ?
- Les ressources pour le français le sont-elles également ?
- Si les ressources pour le français ne le sont pas, quel formalisme utiliser pour créer une ressource complète ? (en s'appuyant éventuellement sur les travaux anglais)
- Comment compléter des ressources existantes, automatiquement ou semi-automatiquement ?

## 1.4 Plan du mémoire

Afin de pouvoir répondre à ces différentes questions, nous avons examiné les différentes ressources portant sur les verbes, pour l'anglais et pour le français, et nous les présentons dans un premier chapitre. Nous détaillons ensuite l'évaluation des ressources en fonction des besoins d'une application d'extraction d'informations. L'avant-dernier chapitre concerne les diverses réalisations menées au cours de ce stage, et nous terminons par l'évaluation de l'extraction ainsi que des ressources qui nous ont permis de la réaliser.

## Chapitre 2

# Aperçu des lexiques prédicatifs pour le traitement des langues

### 2.1 Généralités sur les verbes

Avant de donner un aperçu de différentes ressources verbales existant pour le français et pour l'anglais, nous souhaitons préciser certaines notions et termes qui sont nécessaires à la bonne compréhension de la suite de ce mémoire.

#### 2.1.1 Le prédicat verbal et les arguments

Une phrase non-nominale s'organise autour d'un verbe et, éventuellement, des mots qui interagissent avec celui-ci. Le verbe décrivant une relation avec les mots qui l'entourent est appelé « prédicat verbal », et les mots intervenant dans la relation sont appelés « arguments ».

Selon les grammaires, les arguments peuvent être :

- soit tous mis sur le même plan,
- soit étudiés de manière à différencier les arguments « primaires », des arguments « adjonctifs ».

Par exemple, la phrase « Pierre a acheté un livre au bouquiniste à 18h. » peut être analysée comme suit :

- **Pierre** a acheté **un livre** au **bouquiniste** à **18h** (prédicat acheter) (en gras les arguments)
- **Pierre** a acheté **un livre** au **bouquiniste** à *18h* (prédicat acheter) (en gras les arguments primaires, en italique les adjonctions).

Les arguments primaires sont des arguments considérés indispensables à la relation induite par le prédicat verbal. Le prédicat « Acheter » met en jeu un acheteur, un acheté, un vendeur, et éventuellement un moyen de paiement.. Les autres arguments sont souvent considérés comme des adjonctions, car non nécessaires à la relation induite par ce verbe.

### 2.1.2 Valence, Cadres de Sous-Catégorisation et Structure Argumentale

La valence caractérise le nombre d'arguments qu'un prédicat verbal doit faire intervenir pour que la phrase dont il est le centre soit grammaticalement correcte. Le prédicat verbal « pleuvoir », par exemple, a pour valence 0, c'est-à-dire qu'il ne fait intervenir aucun argument : « *Il pleut* ». Le verbe « caresser » a pour valence 2 : « **Pierre caresse son chien** ».

Le schéma de sous-catégorisation permet quant à lui de définir la nature syntaxique des arguments d'un verbe : les arguments sont-ils introduits par une préposition (*pp*) ? Sont-ils des groupes nominaux simples (*np*) ou des propositions subordonnées conjonctives (*sc*) ? Par exemple, les schémas suivant pourraient être des schémas de sous-catégorisation de « penser » :

[ *np* ] penser [ *sc* ] (Je pense qu'il est coupable.)

[ *np* ] penser [ *pp* ] (Il pense à sa mère.)

Le schéma suivant pourrait caractériser une phrase faisant intervenir le verbe acheter :

[ *np* ] acheter [ *np* ] [ *pp* ] (Jean a acheté un livre au bouquiniste).

La structure argumentale concerne le typage des arguments : à chaque argument est attribué un rôle thématique ou sémantique, et la réunion des informations sur les arguments d'un verbe constitue sa structure argumentale.

### 2.1.3 Rôles thématiques et rôles sémantiques

Dans ce mémoire, nous ferons une différence entre rôles thématiques et rôles sémantiques, les uns correspondant à des rôles thématiques précis, les

autres à des rôles thématiques plus généraux.

Selon les théories utilisées, les arguments d'un verbe peuvent être décrits de différentes manières (cf §2.1.2). Certaines théories utilisent une granularité de description des arguments que nous pourrions utiliser pour décrire les arguments de manière langagière ou cognitive. Ce sont les rôles sémantiques :

- Prédicat « Acheter », Arguments[Acheteur, Objet\_acheté, Vendeur]
- Prédicat « Peindre », Arguments[Peintre, Oeuvre]
- Prédicat « Donner un cours », Arguments[Professeur, étudiant, matière].

D'autres utilisent une granularité de description beaucoup moins importante. Ce niveau de description permet de définir de manière assez générale les arguments d'un verbe, mais ne permet pas, indépendamment du verbe, de comprendre le rôle précis joué par les différents arguments. En utilisant la logique de description mise en place par Patrick Saint-Dizier pour Volem [FSDV<sup>+</sup>02], nous pouvons obtenir les rôles suivants (*voir fig 2.13 pour une description des rôles utilisés*) :

- Prédicat « Acheter », Arguments[Agent+Destinataire, Thème holistique(Argument qui n'est pas affecté dans son intégrité par une action), Source]
- Prédicat « Peindre », Arguments[Agent, Thème incrémental (Argument qui est affecté dans son intégrité par une action)]
- Prédicat « Enseigner », Arguments[Agent+Source (un agent qui est la source d'un transfert), tg (thème général : objet sans spécificités), Destinataire].

Le problème de ces rôles thématiques est qu'ils ne sont pas suffisants en eux-même, mais aussi qu'il peut être difficile d'affecter un rôle thématique à un argument. Par exemple, lors d'un achat, l'objet acheté est-il ou non affecté dans son intégrité ? Le changement de possession d'un objet l'« affecte-t-il » dans son intégrité ?

Un dernier niveau de description des arguments est tellement général que les traits grâce auxquels sont décrits les arguments n'ont aucun rapport avec la relation que les arguments entretiennent avec leur prédicat. C'est le niveau de description utilisé dans les Tables du LADL :

- Prédicat « Acheter », Arguments[Humain, Non-humain|Abstrait, Humain]
- Prédicat « Peindre », Arguments[Humain, Non-humain]
- Prédicat « Enseigner », Arguments[Humain, Abstrait, Humain].

### 2.1.4 Les alternances

Chaque verbe – ou plutôt sens de verbe – exhibe des caractéristiques transformationnelles. Celles-ci correspondent aux différentes façons dont s'agencent dans une phrase un verbe et ses arguments. Elles sont appelées « alternances ». Par exemple, les phrases :

- Jean a acheté un livre au bouquiniste.
- Jean lui a acheté un livre.
- Ce livre a été acheté au bouquiniste par Jean.
- Tout s'achète.

correspondent à autant d'alternances différentes du verbe « Acheter ».

Les différentes alternances que supportent un sens de verbe décrivent l'ensemble des constructions syntaxiques possibles qui ne modifient pas le sens du verbe.

### 2.1.5 Les classes de Levin

L'étude des différentes alternances a donné l'idée à Beth Levin de regrouper les verbes par les alternances qu'ils acceptent ou non, afin de démontrer que les verbes qui partagent les mêmes alternances possèdent des traits sémantiques en commun [Lev93].

Beth Levin classe donc les différents verbes de la langue anglaise dans des classes sémantiques qui encodent les différentes alternances acceptées par les verbes qui les composent.

Les classes ainsi créées contiennent tout de même des membres qui exhibent des traits sémantiques différents. Les classes de Levin ont été enrichies par des classes intersectives, qui ont été créées en groupant ensemble des classes qui partageaient un minimum de trois membres ; ceci a été réalisé en partant de la constatation que beaucoup de verbes sont présents dans plusieurs classes [KS05].

Les classes créées par Levin ne sont donc pas parfaites, loin de là, mais elles permettent quand même de faire apparaître des liens entre la sémantique de certains verbes et leur syntaxe.

## 2.2 Ressources pour l'anglais

### 2.2.1 VerbNet

VerbNet est une ressource développée par Martha Palmer de l'Université de Colorado Boulder. Cette ressource est fondée sur les classes<sup>1</sup> de verbes définies par Beth Levin [Lev93], enrichies par les classes intersectives (cf §2.1.5).

Aujourd'hui, VerbNet compte 4000 entrées lexicales correspondant à autant de sens de verbes, 191 classes de verbes, 52 cadres syntaxiques (équivalences cf § 2.1.4) et 23 rôles thématiques différents.

#### Les rôles thématiques dans VerbNet

Les rôles thématiques dans VerbNet sont des rôles thématiques génériques. M. Palmer s'inspire des travaux de Gruber, Fillmore et Jackendoff pour ses rôles thématiques [KS05], dont les plus communs sont décrits en figure 2.1.

Le principal reproche que l'on puisse faire à ce type de rôles réside dans le fait qu'il n'y a aucun moyen de déterminer ni un nombre précis, ni les types de rôles thématiques qui pourraient servir à caractériser de manière exhaustive les types des arguments des verbes. Il est également difficile de classer certains arguments dans tel ou tel rôle thématique.

#### La polysémie dans VerbNet

Dans VerbNet, les verbes sont regroupés dans des classes sémantiques que M. Palmer appelle « intersective Levin classes ». Un verbe peut faire partie de plusieurs classes à la fois. Un verbe aura autant de sens différents identifiés par VerbNet que de classes différentes auxquelles il appartient. Prenons l'exemple d'un des verbes les plus polysémiques en anglais, « cut » (tab 2.1). Celui-ci appartient à 7 classes différentes. Il a donc 7 sens différents identifiés. Le verbe cut n'a pas pour autant seulement 7 sens différents en anglais, comme en témoigne WordNet qui en recense plus de 20. Certains emplois métaphoriques, notamment ceux du cinéma, du montage, exhibent

---

1. cf §2.1.5

- **Agent** : l’instigateur d’une action.
- **Patient** : un participant à la phrase affecté par une action.
- **Theme** : un participant à la phrase qui est à un endroit, ou qui va d’un endroit à un autre.
- **Expérencier** : un participant à la phrase qui est au courant de quelque chose (par exemple les sujets de verbes comme aimer, admirer).
- **Stimulus** : un évènement ou un objet qui apporte une réponse de type psychologique à un « Expérencier »
- **Instrument** : un objet qui engendre un changement d’état à quelque chose qui rentrerait au contact avec lui.
- **Location** : un participant à la phrase qui exprime un lieu.
- **Source** : un participant à la phrase qui est le point de départ d’un mouvement ou d’un transfert.
- **Goal** : un participant à la phrase qui est le point d’arrivée d’un mouvement ou d’un transfert.
- **Recipient** : un participant à la phrase qui est la cible d’un transfert.
- **Benefactive** : une entité qui bénéficie d’une action (en anglais, c’est le participant à l’alternance : « Benefactive alternation »)

FIGURE 2.1 – Les rôles thématiques les plus communs de VerbNet

les mêmes alternances que les sens propres. VerbNet étant fondé sur une analyse purement syntaxique du comportement des verbes, ceci explique que certains sens de verbes ne soient pas référencés.

La polysémie dans VerbNet est donc gérée, mais de manière non-exhaustive car l’approche de la polysémie des verbes n’est pas fondée sur une approche purement sémantique, mais syntaxique.

### Les alternances dans VerbNet

VerbNet reprend en partie les classes de Levin. Les alternances de VerbNet reprennent donc les variations de diathèse définies par Beth Levin [Lev93]. Chacune des alternances est exemplifiée, et celles-ci sont appelées « cadres syntaxiques » dans VerbNet. Ces alternances sont extrêmement complètes, puisqu’elles mettent même en jeu l’ajout ou non d’un adverbe, comme pour l’alternance « Characteristic Property of Instrument » : « This knife cuts well. ». En revanche, VerbNet ne donne pas d’informations quant à l’utilisation ou non de la forme passive pour un verbe.

classe	cadre	exemple
cut-21.1-1	Transitive	"Carol cut her finger."
cut-21.1	Conative	"Carol cut at the bread with a knife."
cut-21.1	Middle Construction	"The bread cuts easily."
split-23.2	Intransitive (+ source PP)	"The twig broke off the branch."
split-23.2	Apart Reciprocal Alternation Intransitive	"The twig and the branch broke apart."
build-26.1-1	Sum of Money Subject Alternation	"\$100,000 builds a house out of sticks."
build-26.1	Benefactive Alternation (for variant)	"Martha carved a toy for the baby."
amuse-31.1	Transitive (+ with-PP)	"The clown amused the children with his antics."
hurt-40.8.3-1	Unintentional Interpretation of Object	"Tessa hurt/sprained her ankle."
hurt-40.8.3-1-1	Intransitive	"My ankle twisted."
hurt-40.8.3-2	Unintentional Interpretation of Object	"Tessa hurt herself."
braid-41.2.2	Transitive	"Celia brushed her hair."
meander-47.7	There-insertion	"There meanders through the valley a river."

TABLE 2.1 – Le verbe cut dans VerbNet (extraits)

### 2.2.2 PropBank

PropBank [PGK05] est une ressource lexicale syntaxique développée par l'Université de Pennsylvanie. Celle-ci consiste en un corpus annoté de 300.000 mots. Le but de cette ressource est d'aider les développeurs de systèmes d'apprentissage statistique en leur fournissant les données nécessaires.

Voici un exemple de l'annotation PropBank[PGK05] (fig 2.2) :

Dans PropBank, chacun des arguments<sup>2</sup> d'un verbe est annoté. Il peut

---

2. cf §2.1.1

**Forme active :**

[*Arg0* Chuck] *bought*[*Arg1* a car][*Arg2* from Jerry] [*Arg3* for \$1000].

[*Arg0* Jerry] *sold*[*Arg1* a car][*Arg2* to Chuck] [*Arg3* for \$1000].

**Forme passive :**

[*Arg1* A car] was *bought*[*Arg0* by Chuck].

[*Arg1* A car] was *sold*[*Arg0* to Chuck] [*Arg2* by Jerry].

[*Arg2* Chuck] was *sold*[*Arg1* a car] [*Arg0* by Jerry].

FIGURE 2.2 – Annotation PropBank de « buy » et « sell »

y avoir jusqu'à 6 arguments (numérotés de 0 à 5), et les numéros de ceux-ci correspondent à l'importance qu'ils ont au sein de la structure argumentale du verbe auquel ils se rapportent.

L'agent est toujours étiqueté *Arg0*. Il se peut donc, comme montré plus haut dans la phrase passive, qu'*Arg0* ne soit pas à la position du sujet grammatical, mais à la position du complément d'agent pour les phrases passives. Cela peut être également le cas pour les verbes dont le sujet grammatical n'est pas celui qui réalise une action, mais celui qui la cause (*induced action verbs*[Lev93] : *Sylvia jumped the horse over the fence*). Dans ce cas, le sujet grammatical reçoit l'étiquette *ArgA*[SD98].

Le complément d'objet direct, quant à lui, est généralement étiqueté *Arg1*, et le complément d'objet indirect, s'il en existe un, *Arg2*[AMG04]. Il peut exister également d'autres arguments qui sont considérés comme faisant partie intégrante de la structure argumentale d'un verbe : dans le cas de *buy*, le complément circonstanciel de montant « for \$1000 », fait partie de cette structure<sup>3</sup>.

Dans PropBank, l'ensemble des arguments d'un verbe est appelé cadre (*frame*). Il se peut aussi qu'un verbe ait dans sa structure argumentale des arguments qui ne font pas partie de son cadre. Ces arguments sont appelés « adjonctions<sup>4</sup> ». Il en existe de plusieurs types. Ils sont tous étiquetés *ARGM*- suivis de leur description : *DIR* pour le complément circonstanciel de direction, *LOC* pour le complément circonstanciel de lieu...

Des arguments jouant un rôle similaire peuvent recevoir des étiquettes différentes selon le verbe considéré. Voici l'exemple des cadre de *buy* et de *sell* dans PropBank[PGK05] (tab 2.2) :

3. Dans un cas, je parle de syntaxe de surface, dans l'autre de syntaxe profonde. Les deux sont liées dans le sens où les auteurs cités considèrent la construction de base (généralement sujet verbe COD COI) comme base pour l'annotation numérique de leurs arguments.

4. cf §2.1.1

<i>args</i>	<i>buy</i>	<i>sell</i>
Arg0	buyer	seller
Arg1	thing bought	thing sold
Arg2	seller	buyer
Arg3	price paid	price paid
Arg4	benefactive	benefactive

TABLE 2.2 – Les cadres dans PropBank

Cette ressource est extrêmement efficace pour ce qui est de l'étiquetage d'arguments de manière statistique [PGK05]. Le problème est qu'il ne référence pas toutes les structures différentes d'un même verbe, étant donné qu'il n'est pas fondé sur une étude linguistique des verbes, mais sur une étude en corpus.

Un autre point peut sembler gênant : s'il existe des cadres pour les verbes, ces cadres sont spécifiques à chaque verbe. Prenons l'exemple de « sell » et « buy ». Dans les deux phrases :

*Arg0* Chuck *bought*[*Arg1* a car][*Arg2* from Jerry] [*Arg3* for \$1000].  
*Arg0* Jerry *sold*[*Arg1* a car][*Arg2* to Chuck] [*Arg3* for \$1000].

« a car » est défini comme l'Arg1, qui est d'après le cadre, soit *thing sold*, soit *thing bought*. PropBank multiplie donc les rôles sémantiques pour un argument qui joue le même rôle dans deux phrases différentes, chose qui peut s'avérer gênante lors d'une tâche d'extraction.

### 2.2.3 FrameNet

FrameNet est un projet mené par Collin F. Baker, J. Fillmore, et John B. Lowe de l'Université de Berkeley. Le but de ce projet est de construire une ressource lexicale couvrant une grosse partie de la langue anglaise, et contenant des phrases annotées sémantiquement et syntaxiquement. Cette ressource est organisée de manière hiérarchique, et comprend verbes, noms, et adjectifs regroupés dans des cadres (frames). Les entrées lexicales sont au nombre de 9000, dont un peu plus de 6000 sont complètement annotées. Les cadres sont au nombre de 625, et sont exemplifiés par plus de 135.000 phrases annotées.

### Les cadres au sens de FrameNet

Que sont exactement les cadres de FrameNet, sur lesquels sont basés la hiérarchie de cette ressource ?

Il faut tout d'abord savoir que FrameNet est fondé sur la sémantique des cadres [Min]. Ceci implique donc les propriétés suivantes sur les cadres de FrameNet [LBF97] :

Un cadre encode des connaissances sur le monde. Certains cadres encodent des connaissances sur les transactions commerciales, d'autres sur le fonctionnement de la justice... A chaque cadre correspondent des scénarios « stéréotypés » : ce sont des situations dans lesquelles on attend que certains évènements se produisent et que certains états soient atteints. Par exemple, le cadre JUDGMENT encode dans FrameNet qu' « un **arbitre** porte un jugement sur un **évalué**. Ce jugement peut s'avérer négatif ou positif... ».

Un cadre comporte également des « éléments de cadre de noyau » (*core frame elements*). Ces éléments sont les intervenants essentiels d'un scénario du cadre. Dans le cas du cadre JUDGMENT, ces éléments sont :

- l'arbitre
- l'évalué
- *expresser* (la partie du corps ou action d'une partie du corpus par laquelle est établi le jugement fait par l'arbitre)
- la raison (ce qui justifie le jugement de l'arbitre)

En sus de ces éléments de noyau, sont également décrits dans les cadres les éléments qui ne sont pas des éléments de noyau (*non-core frame elements*). Ces éléments ne sont pas essentiels, mais peuvent intervenir dans le cadre qu'ils évoquent.

Les cadres sont organisés de manière hiérarchique, et un cadre de bas niveau hérite de ces prédécesseurs. Par exemple, le cadre **commerce\_good\_transfer** hérite les propriétés du cadre **commercial\_transaction**.

La dernière entrée d'un cadre de FrameNet concerne les entrées lexicales de ce cadre : ce sont les noms, adjectifs et verbes qui évoquent ce cadre. Dans le cadre JUDGMENT, les entrées lexicales sont, entre autres :

- appreciate.v
- appreciation.n
- contempt.n
- deplore.v
- derisive.a
- mockery.n

- reverence.n
- stigmatize.v

### Les rôles sémantiques dans FrameNet

Etant donné que FrameNet est une ressource fondée sur la logique des cadres, chaque cadre possède son propre jeu de rôles sémantiques. Ainsi, les rôles sémantiques, contrairement aux rôles de VerbNet, ne sont pas des rôles génériques, mais des rôles bien spécialisés. Ceci implique donc également que tous les verbes faisant partie d'un même cadre partagent forcément les mêmes rôles. Ceci peut poser aux personnes chargées de construire la ressource un problème de représentation : jusqu'où faut-il aller dans la définition des cadres ?

### Les alternances et arguments de verbe dans FrameNet

FrameNet définit pour chaque verbe une grille de ses différentes réalisations syntaxiques. Celles-ci comportent, pour chacun des éléments du noyau du cadre dans lequel le verbe apparaît, les façons dont ceux-ci agissent au sein des phrases du corpus utilisé par FrameNet (fig 2.3, fig 2.4).

Frame Element	Number Annotated	Realizations(s)
Buyer	(82)	CNI.-- (24) NP.Ext (54) PP[by].Dep (4)
Goods	(82)	DNI.-- (13) NP.Ext (13) NP.Obj (48) Sinterrog.Dep (2) AJP.Dep (2) NP.Dep (4)

FIGURE 2.3 – Les réalisations syntaxiques de buy.v dans FrameNet

On peut constater ici une chose étrange : buy.v apparaît dans le cadre COMMERCE\_BUY, et sell.v apparaît dans le cadre COMMERCE\_SELL. On peut considérer qu'achat ou vente constituent tous deux une transaction commerciale (cadre COMMERCIAL\_TRANSACTION), et qu'ils ont, si l'on suit la logique de constitution de FrameNet, les mêmes éléments de noyau.

Frame Element	Number Annotated	Realizations(s)
Buyer	(64)	INI.-- (50) NP.Obj (2) PP[into].Dep (1) PP[to].Dep (11)
Goods	(64)	DNI.-- (3) INI.-- (1) NP.Ext (31) NP.Obj (27) NP.Dep (2)
Seller	(64)	CNI.-- (28) NP.Ext (32) PP[by].Dep (4)

FIGURE 2.4 – Les réalisations syntaxiques de sell.v dans FrameNet

Or, dans le cas de buy, le vendeur n'est pas un élément de noyau, tandis que dans le cas de sell, l'acheteur est bien un élément de noyau. Ceci peut s'expliquer par le fait qu'en corpus, les ressourceurs ont trouvé moins d'occurrences de buy ayant comme argument un vendeur que d'occurrences de buy sans un vendeur comme argument. Il reste tout de même illogique qu'une ressource fondée sur une approche cognitive de la langue ne définisse pas les mêmes arguments pour les complétifs « vendre » et le verbe « acheter ».

Après les réalisations syntaxiques simples, FrameNet propose des réalisations syntaxiques plus complexes, dans lesquelles on voit plus exactement le rôle que jouent chacun des arguments dans la phrase, et si ces arguments sont oui ou non introduits par une préposition. (cf §2.5).

### Conclusions sur FrameNet

Grâce à la polyvalence de cette ressource, traitant aussi bien de la sémantique que de la syntaxe, il est possible d'identifier le sens d'un verbe par son contexte proche, et d'annoter sémantiquement les arguments qui l'entourent grâce aux différentes constructions syntaxiques repérées en corpus et inscrites dans la base de données FrameNet.

Même si certaines « incohérences » sont présentes, c'est la ressource anglaise qui paraît la plus adaptée à l'extraction d'information.

Number Annotated	Patterns	
82 TOTAL	Buyer	Goods
(3)	CNI --	DNI --
(9)	CNI --	NP Ext
(11)	CNI --	NP Obj
(1)	CNI --	Sinterrog Dep
(2)	NP Ext	AJP Dep
(10)	NP Ext	DNI --
(4)	NP Ext	NP Dep
(37)	NP Ext	NP Obj
(1)	NP Ext	Sinterrog Dep
(4)	PP[by] Dep	NP Ext

FIGURE 2.5 – Le contexte des réalisations syntaxiques de buy.v dans Frame-Net

## 2.3 Ressources pour le Français

Les ressources anglaises sont assez diverses, et offrent une large couverture. Qu'en est-il des ressources françaises ? Aujourd'hui, trois ressources importantes existent, qui traitent des verbes. Il s'agit des Voisins de Le Monde, ressource distributionnelle, des tables du LADL, un lexique-grammaire, et de Volem, une ressource ressemblant sur de nombreux points à VerbNet. Nous avons choisi d'éluder l'étude du DEC. En effet, si la théorie de description établie par Melc'uk paraît être intéressante d'un point de vue qualitatif, le codage sous forme informatisée est tout juste commencé, et aucun des verbes dont nous compterions nous servir pour l'extraction n'est décrit dans la base de données.

### 2.3.1 Les Voisins de Le Monde

Les Voisins de Le Monde est une ressource lexicale distributionnelle construite à partir des articles du Monde de 1991 à 2000. Le corpus du Monde a été analysé par Syntex, un analyseur syntaxique développé par l'ERSS et la société Synomia. Cette ressource permet de mettre en évidence des relations inattendues entre les mots.

L'analyse distributionnelle est fondée sur des relations binaires « prédicat<sup>5</sup>- argument ». La ressource ne traitant que de relations binaires, chacun des arguments est trié selon la relation par laquelle il est lié au verbe (fig 2.6).

Arguments				
Catégorie	Lemme	Relation	Nb cooccurrents	Nb Voisins
A	acheter	—	19	19
V	acheter	auprès de	2	0
V	acheter	avant	1	0
V	acheter	avec	2	0
V	acheter	chez	4	0
V	acheter	dans	24	441
V	acheter	de	10	9
V	acheter	en	32	325
V	acheter	grâce à	1	0
V	acheter	lors de	1	0
V	acheter	obj	471	250
V	acheter	par	1	0
V	acheter	pour	21	173
V	acheter	pour près de	3	0
V	acheter	sous	1	0
V	acheter	subj	120	632
V	acheter	sur	11	31
V	acheter	sur la base de	2	0
V	acheter	via	1	0
V	acheter	à	88	203

FIGURE 2.6 – Liste des relations du lemme acheter dans VDLM

Par la suite, une liste des arguments de chaque entrée « prédicat- relation » est établie (cf §2.7) et comparée aux autres, et la mesure jaccard<sup>6</sup> est appliquée à chacun des couples de « prédicats-relation » partageant des arguments. Ceci permet d'obtenir une mesure de similarité entre prédicats

5. cf §2.1.1

6.  $\frac{m_c}{m_1+m_2-m_c}$  ( $m_c$  : nb d'args en commun,  $m_1$  : nb d'args du prédicat 1,  $m_2$  : nb d'args du prédicat 2)

et de créer une liste de voisins pour chaque prédicat. Cette liste de voisins peut éventuellement rendre compte de similarités entre prédicats, en acceptant le postulat selon lequel le sens d'un mot peut se déterminer d'après son contexte (fig 2.8).

Argument			
Catégorie	Lemme	IM ↑ ↓	Fréquence ↑ ↓
N	hawkeye	8.942	5
S	vieux maison	8.942	5
S	tel titre	8.942	5
S	deux billet	8.942	5
S	quantité important	8.942	5
S	billet de loterie	8.594	12
S	valeur étranger	8.403	7
S	part de fond	8.323	7
S	page de publicité	8.249	5
S	encart publicitaire	8.249	5
S	beau maison	8.154	5
N	patate	8.067	5
S	billet de train	7.814	11
N	saucisson	7.726	8
N	mig	7.521	7

FIGURE 2.7 – Liste des arguments du prédicat acheter obj de VDLM

### 2.3.2 Les Tables du LADL

Les tables du LADL sont un lexique-grammaire établi par Maurice Gross et regroupant 6000 verbes répartis dans des tables construites d'après des similitudes de comportements des verbes qui les composent.

Chaque table du lexique-grammaire contient un certain nombre de propriétés, et des verbes, pour lesquels les propriétés seront validées ou invalidées. Ces propriétés sont des informations sur :

- les réalisations possibles des arguments ;
- les propriétés syntaxiques du verbe ou de ses arguments ;
- les sous-catégorisations alternatives ;
- les possibilités de redistributions (passif long, passif court...)

[GGPF06].

Cat	Lemme	Relation	Nb cooccurents	a	Prox Jaccard ↑ ↓
V	vendre	obj	572	307	0.52
N	achat	de	348	193	0.441
N	vente	de	653	279	0.413
V	fabriquer	obj	280	148	0.344
V	produire	obj	491	172	0.302
N	production	de	510	178	0.3
V	acquérir	obj	352	152	0.289
N	distribution	de	238	116	0.282
N	acquisition	de	211	109	0.28
N	fabrication	de	210	105	0.278
V	destiner	obj	714	189	0.278
V	posséder	obj	620	186	0.272
N	importation	de	144	80	0.258
N	prix	de	1087	247	0.257
N	marché	de	1001	233	0.249

FIGURE 2.8 – Les 15 premiers voisins d’acheter obj extraits de VDLM

### Les réalisations syntaxiques des verbes dans les tables du LADL

Les informations syntaxiques contenues dans les tables du LADL sont très riches. Elles contiennent des informations sur la pronominalisation des verbes, le nombre des arguments, leur introduction par une préposition, le rôle du sujet, l’auxiliaire du verbe, la possibilité ou non de passivisation... Le problème de la représentation réside dans le format de la table (fig 2.9 et fig 2.10). Ce format nécessite une bonne connaissance du lexique-grammaire de Maurice Gross, et son format n’est pas directement exploitable par des applications de TAL [GGPF06]. C’est pour cette raison que le LORIA développe SynLex, des graphes représentant les informations contenues dans le lexique-grammaire.

### Les rôles des arguments dans les tables du LADL

Dans les tables du LADL, les arguments ont certaines propriétés que Maurice Gross qualifie de « traits syntaxiques », plus que d’une réelle caractérisation sémantique [Gro86]. Pour les arguments nominaux, le lexique définit si ce sont des arguments à trait humain ou non. Pour les arguments nominaux, le lexique définit des types (locatifs...). [GGFP05]

	N0 =; Nhum	N0 être V-n	N2 bénéficiaire	N1 =; Nhum	N0 V N0pc	N1 =; N-hum	N1 être V-n	N1 =; Dnum N	N1 =; coup	Vrép =; à	Prép =; de	N2 =; N-hum	Ppv =; lui	Ppv =; y	N2 être V-n	N2 =; Npl obl	N1 est Vpp	N1 =; Nabs
abandonner	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
abouler	+	?	?	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
accepter	+	?	?	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
accorder	+	?	+	+	+	+	?	?	?	?	?	?	?	?	?	?	?	?
acheter	+	?	?	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
acquérir	+	?	?	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
acquitter	+	?	+	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
adjoindre	+	?	+	+	?	+	?	?	?	?	?	?	?	?	?	?	?	?
adjuder	+	+	+	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
administrer	+	+	+	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
adresser	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
adresser	+	?	+	+	?	+	?	?	?	?	?	?	?	?	?	?	?	?
affermer	+	?	+	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
affermer	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
affermer	+	+	?	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
affermer	+	?	?	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
affréter	+	?	?	?	?	+	?	?	?	?	?	?	?	?	?	?	?	?
aliéner	+	?	+	+	?	+	?	?	?	?	?	?	?	?	?	?	?	?
aliéner	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

FIGURE 2.9 – Extrait de la table 36DT des tables du LADL

Les arguments dans les tables du LADL n'ont donc pas de traits sémantiques à proprement parler, puisque mêmes les caractéristiques « humain » ou non sont considérées comme des caractéristiques syntaxiques.

### La logique de constitution des tables du LADL

Chaque table du lexique-grammaire est associée à un cadre de sous-catégorisation de base. Tous les verbes participant à ce cadre sont regroupés dans la table correspondant. Ensuite, certaines propriétés vont venir se greffer à ces verbes, que les verbes acceptent ou non. Cette méthodologie de constitution des tables peut être comparée à la méthodologie de constitution des classes de Beth Levin. Ceci implique que l'on peut tirer certains traits sémantiques de verbes d'après leur appartenance à telle ou telle table.

### La sémantique dans les tables du LADL

En examinant les verbes présents dans chacune des tables, on s'aperçoit que certains verbes qui exhibent des propriétés sémantiques proches, sont regroupés au sein d'une même table. Prenons l'exemple de la table 36DT, dont nous avons un extrait à la figure 2.9 ; sont présents dans cette table :

- acheter

## Table 36 DT (Datif)

Construction de base :  $N_0 V N_1 \text{ à } N_2$ , avec  $N_2 = N_{hum}$ , pronominalisable en *lui*.

[...]

### Exemple : attribuer

- $N_0 = : N_{hum}$  **Paul** attribue un bureau à Marie.
- $N_2$  bénéficiaire **Marie** est bénéficiaire du bureau.
- $N_1 = : N_{hum}$  Paul attribue **une secrétaire** à Luc.
- $N_1 = : N_{-hum}$  Paul attribue **une chaise** à chacun.
- $N_1 = : V_{-n}$  La chaise est une attribution.
- $N_1 = : D_{num} N$  Paul attribue **mille francs** à chacun.
- $N_2 = : N_{-hum}$  Paul attribue un numéro à **chaque sculpture**.
- $P_{pv} = : lui$  Paul **lui** attribue un numéro (à cette sculpture).
- $N_1 = : N_{abst}$  Paul attribue **un travail** à chacun.

FIGURE 2.10 – Etrait de la description de la table 36DT des tables du LADL

- donner
- emprunter
- payer
- prêter
- racheter
- revendre
- vendre
- voler
- ...

Tous les verbes traduisant un transfert de propriété sont présents dans la table 36DT. Ceci rejoint en partie la théorie de Beth Levin [Lev93], selon laquelle les verbes partageant les mêmes traits syntaxiques, partagent également certains traits sémantiques.

La deuxième observation que nous pouvons faire concerne la polysémie. Prenons le verbe « acheter ». Il a un sens propre : « J'ai acheté du beurre. », mais également un sens figuré : « J'ai acheté le geôlier. ». Le verbe « racheter » se retrouve dans deux tables du lexique-grammaire : la table 36DT (*fig 2.9 et fig 2.10*) et la table 32H, dont la caractéristique principale est : « Construction de base :  $N_0 V N_1$  (avec  $N_1 = : N_{hum}$  obligatoire) », ce qui correspond au sens métaphorique du verbe « acheter ».

Prenons maintenant l'exemple du verbe « couper », un verbe extrêmement polysémique. Il est présent dans 16 tables du lexique-grammaire.

Ceci confirme qu'il est possible de tirer des traits sémantiques de l'appartenance d'un verbe à une table donnée.

### Conclusions sur les tables du LADL

Les tables du LADL sont une ressource à large couverture avec une très forte caractérisation syntaxique des verbes et de leurs arguments. En revanche, elle n'est pas directement exploitable par des applications de TAL, mais des travaux sont menés par le LORIA pour adapter format du lexique-grammaire [GGPF06] de manière à le rendre exploitable. Cependant, si cette ressource nous donne des informations très complètes sur le comportement syntaxique des verbes, celle-ci est pratiquement dénuée d'informations sémantiques, que ce soit sur le sens des verbes, ou sur la caractérisation des arguments de ceux-ci.

#### 2.3.3 Volem

Volem est une ressource lexicale multilingue dont les entrées sont des verbes, et dans lesquelles sont décrits leur syntaxe et leur sémantique, ainsi qu'un schéma de sous-catégorisation. Cette ressource décrit à l'heure actuelle 1700 verbes.

#### Description sémantique des entrées lexicales et polysémie dans Volem

Pour décrire le sens des verbes de Volem, une référence ou plusieurs références sont associées au verbe qui sont les liens vers les entrées des différents sens du verbe dans WordNet. Volem devait être une ressource gérant la polysémie [FSDV<sup>+</sup>02] par l'ajout d'entrées pour des sens de verbes différents ; cependant, la version actuellement proposée ne gère la polysémie que par l'ajout de liens vers les différents sens d'un même verbe dans WordNet.

Par exemple, le verbe couper n'a qu'une seule entrée dans Volem, ainsi que tous les autres verbes polysémiques. Les entrées des verbes polysémiques décrivent le sens le plus usuel du verbe, et si un autre sens du même verbe se comporte de la même manière syntaxiquement, alors ce sens est ajouté dans les références WordNet.

### Les variations syntaxiques dans Volem

Dans Volem, la syntaxe joue un rôle très important. Les entrées des verbes comprennent les alternances que ceux-ci acceptent. Celles-ci, contrairement à une ressource comme VerbNet où les alternances sont décrites grâce à la position des arguments dans la phrase, sont décrites seulement par leur nom, chacune des alternances étant accompagnée d'un exemple général. Les alternances de Volem sont au nombre de 30 (fig 2.11). Chacune de ces alternances induit un certain nombre de cadres syntaxiques différents, si l'on considère qu'un changement de position d'un argument, ou l'introduction d'un argument par une autre préposition conduit à un nouveau cadre syntaxique. Par exemple, l'alternance *pas\_etre\_part\_np\_2pp* (forme passive avec comme sujet un groupe nominal, et comme autres arguments deux groupes prépositionnels) peut se traduire par :

1. Cette barre de chocolat a été achetée par Jean à Jacques
2. Cette barre de chocolat a été achetée à Jacques par Jean
3. Cette barre de chocolat a été acheter par Jean auprès de l'épicier Jean.
- ...

Les alternances pour chaque verbe sont très complètes. Mais manque à cette ressource les prépositions qui introduisent habituellement les arguments des verbes, comme dans VerbNet. Ainsi, la caractérisation du cadre syntaxique du verbe « acheter » est décrite dans la figure 2.12 :

Ces alternances signifient que le verbe accepte les transformations suivantes :

1. **Pierre achète une petite voiture à Paul.**
2. Une petite voiture, **ça s'achète.**
3. **Paul s'achète une petite voiture auprès du marchand de jouets.**
4. **La petite voiture a été achetée à Paul par Pierre.**
5. **La petite voiture a été achetée par Pierre.**
6. **Pierre achète une petite voiture.**
7. **Pierre s'achète une petite voiture**
8. **Pierre achète de la coriandre.**
9. **Pierre achète une voiture.**

- anti\_2np : **Le vent** pousse **la porte**.
- anti\_np : **Les résultats** ont changé.
- anti\_np\_pp : **Le bruit** pousse à *l'agressivité*.
- anti\_pr\_np : **La porte** s'ouvrit.
- anti\_pr\_np\_pp : **Marie** se surprend de *ses commentaires*.
- anti\_refl\_np : **Jean** se délasse.
- caus\_2np : **Marie** envoie **un paquet**.
- caus\_2np\_compl : **Marie** nomme **Paul** *responsable des ventes*.
- caus\_2np\_phrase : **Marie** supplie **Paul** de *faire la vaisselle*.
- caus\_2np\_pp : **Paul** a abrité **Marie** *sous son parapluie*.
- caus\_faire\_inf\_np : **Marie** fait chauffer **le lait**.
- caus\_np(plu) : **Marie et Paul** dansent.
- caus\_np\_2pp : **Marie** parlait de *politique avec son professeur*.
- caus\_np\_pp : **Paul** court *vers sa mère*.
- caus\_refl\_pr\_2np : **Jean** se lave **les mains**.
- caus\_refl\_pr\_np : **Jean** se lave.
- caus\_refl\_pr\_np\_pp : **Jean** se regarde *dans une glace*.
- caus\_support\_np : **Jean** porte un jugement.
- inch\_np\_pp : **Le meuble** bascule *sur le sol*.
- pas\_etre\_part\_np : **Ce livre** a été lu.
- pas\_etre\_part\_np\_2pp : **Un livre** a été donné à *Jean par Marie*.
- pas\_etre\_part\_np\_pp : **Ce livre** a été lu *par Marie*.
- rcpr\_pr\_np : **Jean et Marie** s'écrivent.
- refl\_np\_np : **Elle** s'appelle **Farah**.
- refl\_pr\_np\_pp : **Jean** se regarde *dans la glace*.
- state\_2np : **Jean** a **un frère**.
- state\_2np\_pp : **Le bâtiment** abrite **des machines** *de la pluie*.
- imp : Il pleut.

FIGURE 2.11 – Les différentes alternances de Volem (tirées de la doc en ligne de Volem)

### Les rôles thématiques dans Volem

A l'instar de VerbNet, les rôles thématiques de Volem sont des rôles thématiques génériques. Cependant, ceux-ci sont plus détaillés que dans VerbNet car ils peuvent être combinés afin de décrire aux mieux les arguments d'un verbe. Ils sont au nombre de 16 (fig 2.13).

Les rôles thématiques de Volem permettent une description assez précise pour certains arguments, en tout cas peut-être plus précise en certaines occasions que la description des arguments par VerbNet.

caus\_2np\_pp, anti\_pr\_np, anti\_pr\_np\_pp, pas\_etre\_part\_np\_2pp,  
 pas\_etre\_part\_np\_pp, caus\_2np, caus\_refl\_pr\_2np, caus\_np\_pp,  
 caus\_support\_np

FIGURE 2.12 – Les alternances du verbe acheter dans Volem

Catégorie : **ce qui cause une action**

- **Inic(agent)** : Argument initiateur volontaire d'une action.
- **Inic(tc)** : Argument initiateur d'une action dénué de volonté.

Catégorie : **ce qui subit une action**

- **tg** : Argument sans information spécifique à son sujet (thème général).
- **th** : Argument qui n'est pas affecté dans son intégrité par une action (thème holistique)
- **ti** : Argument qui est affecté, positivement ou négativement par une action (thème incrémental).
- **tiv** : thème incrémental victime.
- **tib** : thème incrémental bénéficiaire.

Catégorie : **expression de la localisation**

- **src** : La source d'un mouvement (au sens général du mouvement).
- **dir** : direction d'un mouvement (pas la destination).
- **dest** : destination d'un mouvement.
- **loc** : localisation (thème générique).
- **pos** : position fixe

Catégorie : **autres types d'arguments**

- **amount** : quantité (au sens large -peut être poids, taille...-)
- **mi** : moyens instrumentaux
- **ident** : identification
- **tcons** : thème conséquence

[FSDV+02]

FIGURE 2.13 – Les rôles thématiques dans Volem

Comparons la grille thématique Volem d'acheter et voler (*fig 2.14 et 2.15*) :

Dans le cas du verbe « acheter », la source du transfert est définie comme **src**. En revanche, la source du verbe « voler » (le volé) est définie comme **src,tiv**, eg. la source d'un transfert qui subit l'action en étant affecté par celle-ci de manière négative.

Dans VerbNet, par contre, aucune distinction sémantique n'est faite entre la victime d'un vol (verbe « to rob »), et le vendeur dans une phrase faisant intervenir le verbe acheter (« to buy »). Tous les deux sont définis comme **source**.

GRILLE THEMATIQUE :
[[inic(agent),dest],[th],[src]]

FIGURE 2.14 – Grille thématique du verbe acheter dans Volem

GRILLE THEMATIQUE :
[[inic(agent),dest],[th],[src,tiv]]

FIGURE 2.15 – Grille thématique du verbe voler dans Volem

### Conclusions sur Volem

Volem est une ressource qui représente les verbes de manière assez précise. Le problème de cette ressource réside dans la non-gestion de la polysémie, ainsi que dans sa faible couverture.

### 2.3.4 Conclusions générales sur les ressources françaises

Les ressources françaises sont beaucoup moins diverses que les ressources anglaises, et beaucoup de ces ressources sont incomplètes, soit au niveau du contenu, soit au niveau de la granularité. Il n'y a aucune ressource aussi complètes ou détaillées qu'un FrameNet. Cependant, celles-ci doivent tout de même être exploitables pour une tâche d'extraction, moyennant éventuellement un travail en amont sur les ressources elles-mêmes.



## Chapitre 3

# Choix d'une ressource pour l'extraction

Afin d'évaluer les ressources, tant en termes de qualité que de quantité, nous avons choisi d'implémenter une tâche d'extraction. Pour ceci, deux solutions s'offrent à nous :

- soit fonder notre approche sur une ressource pour le français qui existe déjà,
- soit utiliser les fondements théoriques d'une autre ressource qui n'existe pas pour le français, afin de créer à la main les entrées nécessaires à la tâche d'extraction que nous allons mettre en place.

Cette tâche d'extraction porte sur l'extraction de relations de rachats d'entreprises, et a déjà été étudiée par Thierry Poibeau[Poi03]. Cependant, les ressources utilisées par T. Poibeau ont été définies en fonction de l'application. Certaines études, comme celles réalisées par Moschitti et Giuglea montrent toutefois l'apport d'un cadre descriptif global pour une tâche d'extraction[AMG04]. Nous voulons étudier ici l'apport d'une ressource à large couverture définie par un cadre théorique cohérent, pour le traitement du français.

### 3.1 Description de la méthode d'extraction

Les verbes décrivent des relations entre les arguments qui les entourent. Il est possible de typer ceux-ci sémantiquement, en fonction de la place qu'ils

occupent dans la structure argumentale. Il est aisé de se rendre compte de ce phénomène en intervertissant sujet et objet dans une phrase : « Bull a racheté CP8 » n'a pas le même sens que « CP8 a racheté Bull. ».

La méthode utilisée doit permettre de remplir un tableau comportant les différents acteurs de la situation à extraire, d'après des phrases trouvées en corpus. Par exemple, dans les phrases :

- Bull rachète CP8 à Schlumberger pour 350 millions d'euros.
- CP8 a été racheté par BULL à Schlumberger pour 350 millions d'euros.
- Schlumberger a vendu pour 350 millions d'euros CP8 à Bull.
- CP8, pour 350 millions d'euros, a été vendu à Bull par Schlumberger.
- Bull acquiert CP8 pour 350 millions d'euros auprès de Schlumberger.

il faut pouvoir identifier Bull comme l'acheteur, CP8 comme l'acheté, Schlumberger comme le vendeur, et 350 millions d'euros comme le montant de la transaction.

Rôles sémantiques	Acheteur	Objet_acheté	Vendeur	Montant
Arguments	Bull	CP8	Schlumberger	350 millions d'euros

TABLE 3.1 – Arguments extraits d'une phrase du corpus FUSACQ

Cette structure des arguments à extraire ressemble fortement au cadre de FrameNet COMMERCIAL\_SELL2.2.3.

L'analyse du corpus se fera grâce au logiciel d'analyse de corpus « Unitex », un analyseur comprenant des notions de syntaxe de bas niveau, de typage de syntagmes, et de morphologie. Cet outil implémente les transducteurs à états finis, des dictionnaires simples, composés, des dictionnaires flexionnels ainsi que les lexique-grammaire.

Unitex étant essentiellement un outil exploitant des automates, la méthode d'extraction consistera à décrire les verbes et leurs arguments de manière positionnelle et syntaxique, de manière à créer des automates Unitex permettant leur extraction ainsi que leur annotation sémantique.

1. Sélection de verbes qui se rapportent à la tâche d'extraction.
2. Mise sous forme exploitable des alternances des verbes.
3. Etiquetage sémantique des différents groupes nominaux à extraire.
4. Réalisation de graphes Unitex permettant le repérage des différentes formes à extraire.
5. Passage des graphes Unitex sur le corpus.

FIGURE 3.1 – Les étapes de la génération de patrons d'extraction

## 3.2 Description détaillée pour l'analyse des besoins en ressources

Les besoins en ressources sont fonction de la méthode utilisée pour réaliser l'extraction. La méthode choisie (création semi-automatique d'automates à partir de ressources lexicales) se déroule en plusieurs étapes, chacune nécessitant des besoins particuliers en ressources. La description de ces étapes est établie à la figure 3.1.

Nous allons étudier plus en détail les étapes de la figure 3.1 —exception faite des étapes pratiques (réalisation de graphes et passage des graphes sur le corpus)— afin de pouvoir établir nos besoins en ressource pour la tâche d'extraction.

### 3.2.1 Sélection des verbes

La méthode d'extraction que nous avons choisie est fondée sur les verbes, car ceux-ci déterminent en partie le sens et la structure des phrases dans lesquelles ils apparaissent. Il est donc important de bien choisir ceux-ci afin d'orienter l'extraction d'information sur la bonne voie.

Pour sélectionner les verbes, plusieurs approches ont été envisagées, qui n'ont pas été menées plus avant, puisque ce problème n'est pas au coeur de la problématique de ce stage.

Retenons tout de même que nous avons observé que l'utilisation des dictionnaires de synonymes existants comme le dictionnaire des synonymes

de l'Université de Caen[Plo] ne fournit pas les synonymes et antonymes attendus. Dans ce dictionnaire en effet, chaque verbe se voit attribuer une liste de ses synonymes qui est établie d'après tous les sens différents de ce verbe indépendamment des problèmes de polysémie. Ainsi pour « acheter », sont retenus « acquérir », « s'offrir », mais également « dépraver », « stupendier », qui ne correspondent pas au sens usuel du verbe « acheter ». Le deuxième point négatif concernant l'utilisation d'un tel dictionnaire pour la sélection de verbes qui vont servir à l'extraction d'information est qu'il ne contient que les synonymes totaux, mais pas les verbes qui ont un sens proche, ex. « racheter » pour « acheter ».

Sélectionner des verbes selon un cadre sémantique auquel ils appartiendraient est également une solution envisageable. La sélection par cadres sémantiques correspond à la hiérarchie de FrameNet. Par exemple, si l'on fait une extraction sur les rachats d'entreprise, on peut considérer qu'il s'agit d'une **transaction commerciale**, et récupérer tous les verbes appartenant à ce cadre. Cette ressource n'existe malheureusement pas en français, mais cela serait le cas, les résultats seraient tout de même à compléter.

Pour cela, le moyen qui paraît le plus approprié est une analyse distributionnelle en corpus. En étudiant le contexte de chaque verbe, des relations de proximité sémantique entre les verbes peuvent être dégagées. Plutôt que de réaliser une analyse distributionnelle, ce qui ne correspond pas au contexte de ce stage, nous avons choisi d'examiner les résultats que nous pouvons obtenir grâce à une analyse distributionnelle en partant de la ressource **Les Voisins de Le Monde** (cf §2.3.1). Cependant, il est admis que les résultats obtenus avec une analyse distributionnelle par dérivation d'un verbe jugé représentatif de la tâche d'extraction (ex « racheter » pour une extraction d'information sur les rachats d'entreprises, « condamner » pour une extraction d'information sur les différentes condamnations pour un crime donné...) sont assez bruités.

En partant de la ressource des Voisins de Le Monde, nous avons pu constater ce phénomène. Le tableau 3.2 montre les voisins obtenus lors d'une recherche sur les prédicats acheter suj, racheter suj et revendre à.

Il est sans doute possible d'améliorer ces résultats en combinant les résultats obtenus pour chacun des prédicats. Nous avons pensé à deux approches : la première consiste à simplement fusionner les résultats de différents prédicats représentatifs des relations à extraire. La deuxième consisterait à éliminer les voisins non pertinents des prédicats représentatifs par rapport à leur similarité avec les autres prédicats représentatifs. Cette deuxième approche n'a pas pu être étudiée plus en détails, les Voisins de Le Monde n'étant pas une ressource libre. Après avoir communiqué par

mail avec Didier Bourigault, il semblerait que la meilleure méthode serait la seconde. Voici tout de même les résultats obtenus grâce à la fusion des différents prédicats issus des verbes « acheter », « racheter » et « revendre » présentés au tableau 3.3.

Il semble donc possible d'améliorer la sélection de verbes obtenue par une analyse distributionnelle sur un corpus général par des méthodes combinant les résultats de différents verbes représentatifs du domaine. La question étant secondaire dans ce stage, et les ressources difficiles d'accès, nous partirons d'une liste de verbes définie à la main, .

### 3.2.2 Quel type de schémas de sous-catégorisation ?

L'approche que nous mettons en place (l'extraction d'informations par des patrons d'extraction) nécessite l'acquisition de schémas syntaxiques pour les verbes qui vont servir à l'extraction. Voici deux types de schémas possibles, l'un ressemblant à ce que l'on peut obtenir avec VerbNet, l'autre plus précis, correspondant plus à un schéma FrameNet :

- $Arg_1$  rachète  $Arg_2$  **Prep**  $Arg_3$ .
- $Arg_1$  rachète  $Arg_2$ .
- $Arg_1$  rachète **Prep**  $Arg_3$   $Arg_2$ .
- $Arg_1$  [**Aux : avoir**] racheté  $Arg_2$  **Prep**  $Arg_3$ .
- ...

ou du type :

- *Acheteur* rachète *Objet Achat* **Prep** *Vendeur*
- *Objet Achat* est racheté par *Acheteur*.
- *Vendeur* revend *Objet Achat* **Prep** *Acheteur*

Peuvent éventuellement être ajoutés à ces schémas des arguments — comme des compléments circonstanciels — qui pourraient être définis comme importants dans la structure d'un verbe : ex. complément circonstanciel de moyen ou de montant — « pour 100.000 euros »—. Ce type de schémas peut aisément être intégré à l'analyseur de corpus Unitex. Dans un cas comme dans l'autre, il est nécessaire de disposer de ressources caractérisant au minimum les alternances, et si possible un rôle associé à chaque argument ; il peut s'avérer utile de connaître l'auxiliaire du verbe dans le cas où l'on aurait à extraire des formes utilisant le passé composé, ou autres temps nécessitant un auxiliaire. Les tables du LADL, Volem ainsi que la ressource pour l'anglais FrameNet encodent les différentes structures de

phrase possible. Cependant, seules les Tables du LADL encodent l'auxiliaire des verbes.

### 3.2.3 Etiquetage sémantique

Pour rendre l'extraction possible ou plus précise, il faut identifier les différents groupes de mots qui peuvent correspondre à des arguments à extraire, et leur attribuer un rôle sémantique. Par exemple, lors de l'extraction d'information sur les rachats d'entreprises, il faudrait annoter les groupes de mots qui peuvent correspondre à un acheteur, à une entreprise ou partie d'entreprise, et ceux pouvant correspondre à un vendeur.

L'étiquetage sémantique de textes est assez complexe. Ceci n'étant pas la priorité de l'étude, l'étape d'annotation a été réalisée de manière ad hoc, et sera décrite plus avant dans le chapitre **Réalisations**.

## 3.3 Une nouvelle ressource pour le français, Volem ou les tables du LADL ?

Une fois les besoins en ressource analysés, il faut passer au choix concernant une ressource. FrameNet remplit tous les besoins en ressources que la méthode d'extraction implique. Aucune ressource de ce type n'est cependant disponible pour le français. Se pose alors la question de savoir s'il est nécessaire de disposer d'une nouvelle ressource pour le français, ou si les ressources existantes, moyennant éventuellement quelques ajouts, sont suffisantes à une tâche d'extraction.

### 3.3.1 Vers une nouvelle ressource plus adaptée à l'extraction d'information ?

FrameNet est une ressource utilisant la théorie des cadres (situationnels). Elle est par conséquent adaptée à de nombreuses tâches d'extraction, puisque les entités à extraire sont souvent les acteurs d'une « situation ». Il convient toutefois de noter que toutes les tâches d'extraction ne sont pas forcément décrites dans une telle ressource. En effet, les développeurs d'une ressource se doivent de faire un choix de niveau de description. Il s'avère que dans le FrameNet anglais, le cadre le plus proche de la tâche d'extraction à

réaliser (eg. les rachats d'entreprise), est COMMERCIAL\_TRANSACTION. Ce cadre n'encode toutefois pas toutes les entrées lexicales spécifiques aux rachats d'entreprise (fusions, offres publiques de rachats...), puisqu'il réunit les termes en rapport avec les transactions commerciales au sens général.

Même si nous disposions pour le français d'une ressource du même type que FrameNet, il y aurait des retouches à faire, tant au point de vue de la couverture de la ressource que de certains choix faits par les concepteurs de la ressource (cf 2.2.3). Cependant, une telle ressource nous permet de définir aisément un cadre qui faciliterait la suite du travail. La conception d'une telle ressource (et même seulement la création d'un cadre pour l'extraction) nécessiterait un travail énorme, et rien n'est sûr quant à la création future d'un FrameNet pour le français. Autant donc tenter de se servir des ressources déjà existantes, plutôt que de fonder le travail à suivre sur le développement d'une ressource qui, de toutes manières, ne pourrait pas être parfaitement adaptée au problème de l'extraction d'information, puisqu'elle ne serait pas adaptée à tous les problèmes d'extraction.

### 3.3.2 Volem ou les Tables du LADL ?

Il nous faut maintenant faire un choix : nous savons qu'aucune des ressources, de Volem ou des Tables du LADL n'est réellement adaptée à la tâche d'extraction. Celles-ci devront être modifiées pour être exploitables. Allons-nous partir du contenu des Tables du LADL ou de celui de Volem ?

#### La gestion des alternances dans Volem et les Tables du LADL

Le choix d'une ressource ou de l'autre se fera selon la difficulté éventuelle pour la compléter. En ce qui concerne les schémas de sous-catégorisation, les deux ressources se valent (excepté le fait que les prépositions sont clairement explicitées dans les Tables du LADL, mais pas dans Volem). Reprenons la table 36DT (fig 3.2), et regardons les réalisations syntaxiques qu'elle accepte pour le verbe « acheter ».

Les transformations ou propriétés qui s'appliquent au verbe « acheter » sont :

- N0 = : Nhum (N0 humain)
- N1 = : N-hum (N1 non-humain)
- N1 être V-n (transformation)

	N0 =: Nhum	N0 être V-n	N2 bénéficiaire	N1 =: Nhum	N0 V N0pc	N1 =: N-hum	N1 être V-n	N1 =: Dnum N	N1 =: coup	Prép =: à	Prép =: de	N2 =: N-hum	Ppv =: lui	Ppv =: y	N2 être V-n	N2 =: Npl obl	N1 est Vpp	N1 =: Nabs
abandonner	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
abouler	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
accepter	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
accorder	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
acheter	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
acquérir	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
acquitter	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
adjoindre	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
adjuder	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
administrer	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
adresser	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
adresser	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
affermer	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
affermer	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
affermer	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
affréter	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
aliéner	+	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
aliéner	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

FIGURE 3.2 – Extrait de la table 36DT des tables du LADL (bis)

- Prep = : à (La préposition utilisée est à)
- N1 = : Nabs (N1 peut avoir un trait abstrait)
- N0 V N1 Dnum N (transformation)
- N0 V N1 à N2 Advp (transformation)

Ceci signifie que l'argument N0 est un argument à trait humain, l'argument N1 un argument à trait non-humain, et que celui-ci peut également prendre un trait abstrait (un travail, un droit...).

Le verbe acheter accepte donc, selon les Tables du LADL, les phrases suivantes :

- Jean achète un livre à Pierre. (Construction de base : N0 V N1 à N2)
- Le livre est acheté. (N1 être V-n)
- Jean achète un livre 10 euros à Pierre.
- Jean achète un livre à Pierre.
- Jean lui achète un livre (à Pierre).

La première constatation est qu'une seule forme passive pour le verbe « acheter » est répertoriée dans les Tables du LADL. Ne sont pas indiquées, par exemple, les constructions du type : « Ce livre a été acheté par Pierre à Jean. ». Ensuite, la préposition qui introduit N2 dans les Tables du LADL est « à ». Cependant, il existe beaucoup de cas dans lesquels il est préférable d'utiliser la préposition « auprès de ». En effet, il est plus courant de dire : « Jean a acheté ce livre auprès du marchand. » que de dire : « Jean a acheté

ce livre au marchand ».

En comparaison, voici les alternances définies dans Volem pour le verbe « acheter » (fig 3.3) ainsi que les phrases correspondant :

caus\_2np\_pp, anti\_pr\_np, anti\_pr\_np\_pp, pas\_etre\_part\_np\_2pp,  
pas\_etre\_part\_np\_pp, caus\_2np, caus\_refl\_pr\_2np, caus\_np\_pp,  
caus\_support\_np (cf §2.3.3)

FIGURE 3.3 – Les alternances du verbe acheter dans Volem (bis)

- caus\_2np\_pp : **Jean** a acheté **un livre** à *Pierre*.
- anti\_pr\_np : Jean **lui** achète *un livre*.
- anti\_pr\_np\_pp : Jean **lui** a acheté **un livre**.
- pas\_etre\_part\_np\_2pp : **Ce livre** a été acheté par *Jean* à *Pierre*.
- pas\_etre\_part\_np\_pp : **Ce livre** a été acheté par *Jean*.
- caus\_2np : **Jean** a acheté un **livre**.
- caus\_refl\_pr\_2np : *Jean* **s'**achète *un livre*.
- caus\_np\_pp : **Jean** achète de *la confiture*.
- caus\_support\_np : Jean **fait ses achats**.

Les alternances sont définies de manière plus précise dans Volem que dans les Tables du LADL. Seuls les auxiliaires ne sont pas encodés, ainsi que les prépositions qui introduisent les *pp*.

### La gestion de la polysémie dans les deux ressources

Nous l'avons vu plus tôt, un même verbe dans les Tables du LADL peut appartenir à plusieurs tables. Ceci signifie que ce verbe admet plusieurs groupes de transformations différentes, et il apparaît en pratique qu'il a plusieurs sens différents répertoriés. Dans Volem, la polysémie était sensée être gérée. Nous pouvons d'ailleurs le constater en voyant qu'à chaque entrée de verbe, des sens WordNet[MBF<sup>+</sup>90] sont associés. Cependant, il n'existe qu'une seule entrée pour chaque verbe, correspondant au sens usuel. Les prochaines versions en ligne de Volem devraient gérer de manière plus approfondie la polysémie (communication personnelle avec Patrick Saint-Dizier).

Bien que la polysémie ne soit pas gérée dans Volem, nous pouvons tenter des points de comparaison entre Volem et les Tables du LADL afin de voir laquelle de ces deux ressources est la plus adaptée à la gestion de la polysémie.

### Volem et les Tables du LADL face à la sémantique des verbes

Pour réaliser une extraction précise, il faut pouvoir distinguer les différents sens d'un même verbe. En corpus, il existe deux façons de faire cela :

- soit nous partons de la syntaxe, en considérant que le sens d'un verbe est défini par les alternances qu'il accepte,
- soit nous considérons la sémantique, en acceptant le fait que le sens d'un verbe est défini par son contexte, et que par conséquent, le type sémantique des arguments de ce verbe influe sur son sens.

La première hypothèse atteint ses limites lors de l'étude en corpus. En effet, les verbes polysémiques acceptent, selon leurs sens, des alternances différentes, mais ils ont également des alternances —généralement les plus répandues— en commun. Considérons par exemple le verbe « larguer ». Nous pouvons considérer que selon que l'argument est « les amarres » ou « le coffre de vivres », la phrase aura un sens totalement différent. Considérons également que la phrase « Les amarres ont été larguées par son mari » (sens métaphorique dans le sens où « son mari a largué les amarres » signifie qu'il est parti, mais ne fait nullement intervenir bateau ou amarres) ne se dit pas, mais qu'en revanche la phrase « Le coffre de vivres a été largué par le mousse » est correcte ; il est possible de différencier le sens métaphorique que l'emploi de « les amarres » en tant qu'argument induit, du sens usuel de « larguer » par le fait que le premier sens n'accepte pas la forme passive, tandis que le deuxième l'accepte. Il en va de même pour :

- « Il a pris la porte. »
- « Il a pris un croissant-beurre. »
- « Il a pris la mouche. »
- « Il a pris son cartable »

Certaines de ces phrases acceptent des alternances que d'autres n'acceptent pas : les trois premières par exemple, n'acceptent que l'alternance *caus\_2np*. La quatrième, par contre, qui correspond au sens usuel du verbe « prendre », accepte les alternances : *anti\_2np*, *caus\_2np-pp* (Il a pris son cartable à son fils), *pas\_etre\_part\_np\_pp* (Le cartable de Martin a été pris par son père), et d'autres encore. Il est donc possible de différencier les sens d'un même verbe selon les alternances qu'ils proposent. Mais qu'en est-il quand l'alternance exhibée en corpus semble être la même ? (« Il a pris la porte » vs « Il a pris son cartable ») Comment différencier en corpus sans analyser les arguments, les sens du verbe dans ces deux phrases ? Sans analyse des arguments, l'analyse syntaxique seule ne suffit pas.

Nous partons donc du principe qu'il est possible de restreindre les sens possibles d'un verbe selon l'alternance à laquelle il participe dans une phrase, mais qu'il est nécessaire pour obtenir le sens précis du verbe, de procéder à une analyse de ses différents arguments.

Il est donc nécessaire, dans une base lexicale sur les verbes, d'associer à chaque argument d'un sens de verbe une étiquette sémantique. Les Tables du LADL ont donc un gros inconvénient : le seul typage des arguments est un typage très générique, tandis que le typage des arguments dans Volem est beaucoup plus spécifique.

### Conclusions

Les deux ressources proposent des alternances assez proches, plus précises dans Volem que dans les Tables du LADL. Volem propose également des rôles thématiques plus précis que ceux des Tables du LADL, étant donné qu'il n'y a pas réellement de rôles thématiques dans les tables du LADL (cf 2.3.2). Nous tenterons une approche permettant de relier les rôles thématiques de Volem à des rôles sémantiques nécessaires pour l'extraction. Les prépositions sont gérées dans les Tables du LADL, mais pas de manière assez poussée (des prépositions « usuelles » ne sont pas référencées). Les auxiliaires de chaque verbe sont également encodés dans les Tables du LADL. Dans Volem, ce n'est ni le cas pour les auxiliaires, ni pour les prépositions. Ceci n'est à priori pas gênant, et nous proposerons dans le chapitre suivant des méthodes pour acquérir en corpus les auxiliaires ainsi que les prépositions.

Nous avons donc choisi de travailler avec la ressource Volem, tout en sachant qu'il faudra la compléter, et en espérant pouvoir profiter un jour d'une version de cette ressource qui gère la polysémie.

<b>acheter</b>			<b>racheter</b>			<b>revendre</b>		
	subj			subj			à	
payer	subj	0.418	acquérir	subj	0.291	racheter	à	0.356
intéresser	obj	0.378	consentir	subj	0.242	rivaliser	subj	0.346
préférer	subj	0.375	détenir	subj	0.239	louer	à	0.323
inciter	obj	0.372	vendre	subj	0.232	acheter	à	0.323
posséder	subj	0.350	céder	subj	0.229	miser	subj	0.294
acquérir	subj	0.344	fabriquer	subj	0.229	faire appel	subj	0.276
vendre	à	0.340	contrôler	subj	0.226	acheter	subj	0.254
obliger	obj	0.339	investir	subj	0.224	acquérir	subj	0.250
investir	subj	0.335	posséder	subj	0.224	facturer	à	0.243
percevoir	subj	0.332	développer	subj	0.219	louer	subj	0.240
se intéresser	subj	0.326	conclure	avec	0.213	se lancer	subj	0.237
vendre	subj	0.321	acheter	subj	0.212	contracter	subj	0.235
rechercher	subj	0.317	envisager	subj	0.206	causer	à	0.233
avoir besoin	subj	0.317	se apprêter	subj	0.204	se associer	avec	0.227
dépenser	subj	0.310	négociier	subj	0.200	se associer	subj	0.223
fournir	à	0.310	employer	subj	0.199	se partager	subj	0.217
pratiquer	subj	0.305	construire	subj	0.189	se implanter	subj	0.212
hésiter	subj	0.299	verser	subj	0.184	retirer	subj	0.211
contraindre	obj	0.299	racheter	à	0.180	avoir recours	subj	0.209
choisir	subj	0.299	signer	avec	0.180	anticiper	subj	0.208
verser	subj	0.297	détenir	dans	0.179	projeter		0.207
apprécier	subj	0.297	commercialiser	subj	0.179	récupérer	subj	0.206
proposer	à	0.290	espérer	subj	0.178	devenir	pour	0.204
éprouver	subj	0.285	céder	à	0.177	souscrire	subj	0.203
offrir	à	0.285	afficher	subj	0.176	économiser	subj	0.201
se voir	subj	0.284	conserver	subj	0.172	procurer	à	0.200
séduire	obj	0.284	payer	subj	0.169	exploiter	subj	0.197
garantir	à	0.283	se intéresser	subj	0.168	octroyer	à	0.193
se contenter	subj	0.281	multiplier	subj	0.167	représenter	pour	0.191
se engager	subj	0.281	fusionner	avec	0.164	réserver	subj	0.191
ignorer	subj	0.280	se engager	subj	0.163	rechercher	subj	0.189
prêter	subj	0.280	contraindre	obj	0.159	supporter	subj	0.189
chercher	subj	0.280	financer	subj	0.159	sponsoriser	subj	0.187
supporter	subj	0.280	salarier	de	0.158	garantir	à	0.187
priver	obj	0.279	abandonner	subj	0.157	remplir	subj	0.182
profiter	subj	0.279	exercer	subj	0.157	lancer	auprès de	0.181
consentir	subj	0.275	privatiser	obj	0.156	répartir	subj	0.179
se sentir	subj	0.274	gérer	subj	0.156	se comporter	subj	0.177
répartir	entre	0.274	pratiquer	subj	0.156	endetter	obj	0.175
interdire	à	0.272	intéresser	obj	0.154	transférer	subj	0.173
utiliser	subj	0.272	exploiter	subj	0.153	convoiter	subj	0.173
aider	obj	0.272	réduire	subj	0.153	se tourner	subj	0.173
disposer	subj	0.271	convoiter	subj	0.152	réduire	subj	0.172
fournir	subj	0.271	coûter	à	0.151	vendre	subj	0.171
regarder	subj	0.271	vendre	à	0.151	intégrer	subj	0.171
se lancer	subj	0.269	négociier	avec	0.151	mettre en vente	subj	0.170
conserver	subj	0.269	souscrire	subj	0.149	solliciter	subj	0.170
négociier	avec	0.268	se livrer	subj	0.148	dépenser	subj	0.169
se mettre	subj	0.268	livrer	subj	0.148	consentir	à	0.168
assumer	subj	0.267	réclamer	à	0.147	se sortir	subj	0.168

TABLE 3.2 – Voisins de « acheter subj », « racheter subj » et « revendre à » dans VDLM

louer	1.290000
acheter	1.113000
mettre en vente	1.030000
racheter	0.905000
fusionner	0.900000
céder	0.853000
acquérir	0.826000
recupérer	0.815000
vendre	0.797000
se associer	0.632000
répartir	0.613000
détenir	0.573000
souscrire	0.520000
posséder	0.519000
nationaliser	0.519000
se partager	0.503000
transférer	0.485000
convoiter	0.458000
octroyer	0.413000
intervenir	0.404000
privatiser	0.357000
revendre	0.356000
retirer	0.344000
payer	0.328000
contrôler	0.226000
investir	0.224000
déposséder	0.222000

TABLE 3.3 – Sélection de verbes grâce à la fusion de mesures de similarité des voisins de prédicats de VDLM



## Chapitre 4

# Réalisations

Nous étudierons dans ce chapitre les diverses réalisations menées au cours de ce stage. Les réalisations présentées ici concernent les quatre parties suivantes :

- Le codage des informations de Volem sous une forme plus appropriée,
- L’enrichissement des descriptions des verbes de Volem,
- Les automates permettant l’extraction d’informations,
- La constitution de corpus.

### 4.1 De Volem à une ressource utilisable pour l’extraction

Comme nous l’avons vu dans le chapitre 3.3.2, Volem est une ressource incomplète pour la tâche que nous avons à réaliser. Nous avons donc mis en place des méthodes permettant de la compléter ; ceci passe par l’ajout des auxiliaires, des rôles sémantiques et des prépositions. Nous proposons ici des méthodes pour extraire des corpus les ressources manquant à Volem.

#### 4.1.1 Le codage de Volem sous une forme plus appropriée à la génération automatique de patrons d’extraction

La base de données de Volem est accessible sur le site de l’IRIT : <http://www.irit.fr>. Le seul format de la base de données accessible est celui

que l'on peut voir en interrogeant la base de données sur un verbe (fig 4.1).

<b>Description du verbe : acheter</b>	
<b>GRILLE THEMATIQUE :</b>	
	[[inic(agent),dest],[th],[src]]
<b>LCS :</b>	
<b>ALTERNANCES :</b>	
	caus_2np_pp , anti_pr_np , anti_pr_np_pp , pas_etre_part_np_2pp , pas_etre_part_np_pp , caus_2np , caus_refl_pr_2np , caus_np_pp , caus_support_np
<b>WN :</b>	
	[13,2,3], [13,3,1] , [13,3,8]
<b>EXEMPLE :</b>	
	Il a acheté ce livre à un brocanteur

FIGURE 4.1 – Exemple d'une entrée du lexique de Volem avec le verbe « Acheter »

Ce format n'étant pas idéal pour une automatisation de la tâche — la structure n'est pas adéquate, les données sont sous forme textuelle, les auxiliaires ainsi que les prépositions sont manquants...—, nous avons choisi de coder les verbes que nous avons sélectionnés dans un format plus appropriée; Etant donné que nous avons choisi de travailler avec Unitex, le format dans lequel nous avons codé les informations de Volem, et les informations complémentaires est le format utilisé par Unitex, à savoir un tableau de la même forme que les tables du lexique-grammaire<sup>1</sup>.

Nous avons deux types d'informations à coder dans notre table de données :

1. Les informations données par Volem
2. Les informations que nous rajoutons à Volem

---

1. cf §2.3.2

### Le codage des informations de volem

Il y a trois informations de Volem qu'il nous faut coder. Elles concernent :

1. les alternances,
2. la grille thématique,
3. le sens du verbe.

Les alternances sont codées exactement comme sont codées les transformations du lexique-grammaire(cf §2.3.2). Chaque alternance définie dans Volem est notée dans une colonne, et chaque verbe sélectionné constitue une ligne. A l'intersection d'une ligne (verbe) et d'une colonne (alternance), nous mettons un + si l'alternance en question est valable pour le verbe, un - dans le cas contraire (fig 4.2).

La grille thématique ne fait pas l'objet du même type de codage. Les rôles thématiques de chacun des trois arguments est codé dans une colonne. Ceux-ci sont codés de manière à ce qu'il n'y ait pas de conflit avec Unitex. Par exemple, le rôle thématique [*inic(agent),dest*] sera codé : *inicagent\_dest* (fig 4.2).

Le sens du verbe est noté exactement comme dans Volem ; c'est le sens WordNet, codé comme on peut le voir dans la figure 4.1.

### Le codage des enrichissements apportés à Volem pour l'extraction

A des fins d'extraction, nous devons apporter à Volem des enrichissements, comme décrit en §3.2. Ces enrichissements sont de quatre types :

1. Les auxiliaires,
2. les prépositions,
3. le nom rattaché au verbe,
4. un exemple.

Les auxiliaires sont codés comme les alternances : un + ou - est placé à l'intersection de la colonne codant l'auxiliaire « avoir » ou de celle codant l'auxiliaire « être » avec une ligne codant les propriétés d'un verbe (fig 4.2).

Les prépositions pouvant introduire un argument d'un verbe sont codées au format Unitex. Cela signifie que si un argument accepte la préposition « à » et la proposition « auprès de », alors on écrira dans la case correspondante : `<à>+<auprès> <de>`.

Volem introduit des alternances correspondant à un verbe support suivi de la forme nominale d'un verbe. Par exemple, « Marie réalise un achat ». Cependant, Volem n'encode pas la forme nominale d'un verbe. Celle-ci est donc rajoutée à notre table de données (nous n'avons pas réfléchi à une façon de l'acquérir de manière automatique). La colonne `verbe_support` indique si, en sus de l'alternance `verbe_support` de base, un verbe peut admettre un verbe support suivi d'une forme nominale qui admet la même construction syntaxique que le verbe lui-même.

La dernière colonne de notre table de données est une phrase qui donne un exemple de l'emploi du sens d'un verbe, avec la construction de base de ce verbe si possible (*fig 4.2*).

#### 4.1.2 L'auxiliaire

L'ajout d'un auxiliaire utilise une méthode simple : il s'agit d'étudier, en corpus, si « avoir » apparaît au moins une fois comme auxiliaire d'un verbe. Si tel est le cas, nous pouvons considérer qu'« avoir » est auxiliaire de ce verbe. Sinon, l'auxiliaire est « être ». En effet, les formes passives utilisent uniquement le verbe être. Il est donc possible, si l'auxiliaire d'un verbe est « avoir », que ce verbe soit tout de même précédé d'être :

- CPI a racheté hier Fulmar pour 50 millions d'euros.
- CPI sera éventuellement racheté par Fulmar pour 50 millions d'euros.

Nous avons donc mis en place un automate qui permet, pour chacun des verbes du lexique-grammaire proposé en 4.1.1 de repérer dans une phrase —et ce malgré l'ajout de divers adverbe entre un auxiliaire et le verbe— les auxiliaires qui apparaissent dans le corpus 4.3.

Pour utiliser cette méthode, il est nécessaire de disposer d'un corpus assez large pour qu'y apparaissent au moins une forme active à un temps composé de chacun des verbes, de manière à pouvoir identifier l'auxiliaire des différents verbes.

La deuxième étape de la méthode d'ajout des auxiliaires au lexique-grammaire consiste à étudier les résultats renvoyés par l'automate, et d'ajouter un + à la colonne **être** si le verbe n'a pas admis une seule fois « avoir » comme auxiliaire, ou un + à la colonne **avoir** dans le cas contraire.

Verbe	Etre	Avoir	anti_2np	anti_np	anti_np_pp	anti_pr_np
racheter	-	+	-	-	-	+
revendre	-	+	-	-	-	-
se partager	+	-	-	-	-	-
acheter	-	+	-	-	-	+
fusionner	-	+	-	-	+	-
fusionner	-	+	-	-	-	-
céder	-	+	-	-	-	+
vendre	-	+	-	-	-	+
acquérir	-	+	-	-	-	+
offrir	-	+	-	-	-	-
offrir	-	+	-	-	-	+
reprendre	-	+	-	-	-	+
détenir	-	+	-	-	-	-

[...]

Verbe	state_2	state_2_imp	Th1	Th2	Th3	Prep1	
racheter	-	+	-	inicagent_dest	th	src	<E>
revendre	-	-	-	inicagent_src	th	dest	<E>
se partager	-	-	-	inicagent	tib	<E>	<E>
acheter	-	-	-	inicagent_dest	th	src	<E>
fusionner	-	+	-	inicagent	ti	ti	<E>
fusionner	-	-	-	inicagent	tib	<E>	<E>
céder	-	+	-	inicagent_src	th	dest	<E>
vendre	-	-	-	inicagent_src	th	dest	<E>
acquérir	-	-	-	inicagent_dest	th	src	<E>
offrir	-	-	-	inicagent	tg	dest	<E>
offrir	-	+	-	inicagent_dest	tg	th	<E>
reprendre	-	-	-	inicagent_dest	th	src	<E>
détenir	-	-	-	inicagent	tg	<E>	-

Verbe	Prep2	Prep3	sens	verbe_support	nom
racheter	<E>	<à>+<au>+<aup>	1+		rachat
revendre	<E>	<à>	1+		revente
se partager	<E>	<E>	1-		-
acheter	<E>	<à>+<au>+<aup>	1+		achat
fusionner	<avec>	<avec>+<et>	1+		fusion
fusionner	<avec>	<E>	2+		fusion
céder	<E>	<à>+<au>	1+		cession
vendre	<E>	<à>+<au>+<aup>	1+		vente
acquérir	<E>	<à>+<au>+<aup>	1+		acquisition
offrir	<E>	<E>	2-		-
offrir	<E>	<pour>+en échange	1+		offre
reprendre	<E>	<à>+<au>	1+		reprise
détenir	-	-	1-		-

FIGURE 4.2 – Extraits de la table de données

D'autres méthodes sont envisageables, comme l'utilisation de ressources lexicales informatisées déjà existantes et dans un format pouvant permettre une automatisation de cette tâche.

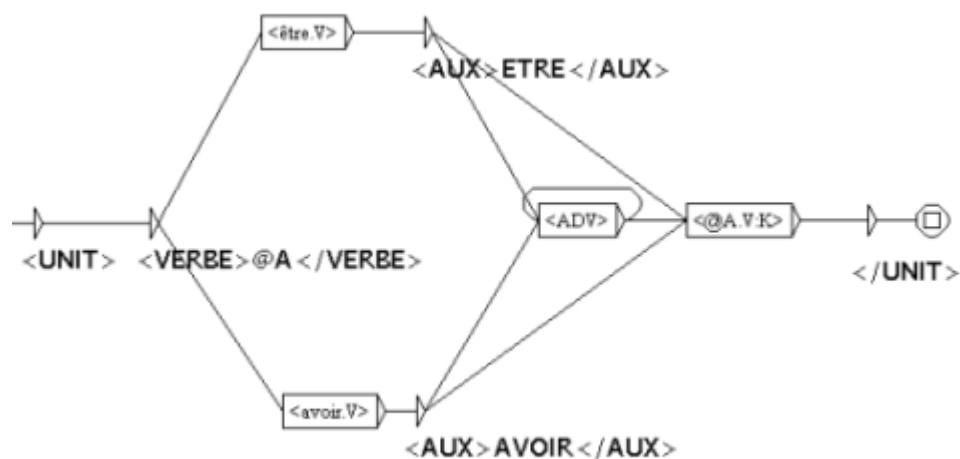


FIGURE 4.3 – L’automate de repérage des auxiliaires

### 4.1.3 Des rôles thématiques aux rôles sémantiques

Les rôles utilisés par Volem sont des rôles thématiques. Afin de réaliser une extraction d’informations comme décrite dans le chapitre 3.1, et également dans un souci de lisibilité du lexique-grammaire 4.1.1, nous voulons passer de la description des rôles argumentaux de Volem à une description des rôles argumentaux plus précise, comme dans FrameNet. Nous avons constaté, après avoir sélectionné des verbes issus de mêmes champs lexicaux, que les rôles thématiques encodés correspondent chacun à un rôle sémantique ; un rôle thématique qui définit les arguments de plusieurs verbes différents n’a une correspondance qu’avec un seul rôle sémantique. Il existe donc, au sein d’un même champ lexical une bijection entre rôles thématiques et rôles sémantiques.

Par exemple, le verbe « peindre » a comme grille thématique : [[inic(agent)], [ti], [pos]]. Le verbe « dessiner », [[inic(agent)], [ti], [pos]], et le verbe créer : [[inic(agent)], [ti]]. [pos] peut être interprété comme le *support* : sur un cahier, sur une feuille de papier... [ti] est l’objet de l’oeuvre, et [inic(agent)], le créateur de cette oeuvre.

Nous avons pu faire la même constatation pour les verbes que nous avons sélectionnés pour la tâche d’extraction sur les rachats d’entreprise, à savoir :

- Racheter [[inic(agent),dest],[th],[src]]<sup>2</sup>
- Revendre [[inic(agent),src],[tg],[dest]]
- Acheter [[inic(agent),dest],[th],[src]]
- Vendre [[inic(agent),src],[th],[dest]]
- Reprendre [[inic(agent),dest],[th],[src]]
- Acquérir [[inic(agent),dest],[th],[src]]
- Céder [[inic(agent),src],[th],[dest]]
- Fusionner [[inic(agent)],[ti],[acp]]
- Offrir [[inic(agent),dest],[tg],[dest]](de l'argent en compensation de quelquechose)
- S'offrir [[inic(agent),src],[th],[dest,tib]](quelquechose)
- Détenir [[inic(agent)],[ti]]

Il est possible d'effectuer une bijection entre rôles thématiques et rôles sémantiques pour les arguments de tous les verbes partageant un même schéma argumental. Ce travail a été réalisé à la main selon un algorithme qui, par manque de temps, n'a pas été codé. Voici les différents rôles sémantiques obtenus avec l'algorithme décrit dans la figure 4.4.

- Racheter [Acheteur,Objet acheté, Vendeur]
- Revendre [Vendeur, Objet acheté, Acheteur]
- Acheter [Acheteur,Objet acheté, Vendeur]
- Vendre [Vendeur, Objet acheté, Acheteur]
- Reprendre [Acheteur,Objet acheté, Vendeur]
- Acquérir [Acheteur,Objet acheté, Vendeur]
- Céder [Vendeur, Objet acheté, Acheteur]
- Fusionner [Fusionné, Fusionné]
- Offrir [Acheteur, Moyen de paiement, Objet acheté]
- S'offrir [Acheteur, Objet acheté]
- Détenir [Détenteur, Objet détenu]

#### 4.1.4 Les prépositions

Nous avons mis en place une méthode qui permet de déterminer les différentes prépositions introduisant les arguments du verbe. Pour ce faire, nous avons réalisé des automates récupérant les mots (si possible prépositions) qui sont positionnés avant une entité reconnue comme un argument éventuel. Les prépositions récupérées sont ajoutées à la structure XML renvoyée par les automates qui sont générés par la table de données. Ces automates seront détaillés dans le chapitre suivant.

---

2. se référer à la figure 2.13 pour une description précise des rôles thématiques de Volem

```

T : Nouveau tableau de structures argumentales
Pour tous les verbes sélectionnés
  Si la structure argumentale n'est pas dans T
    l'ajouter à T
    Demander à l'utilisateur les correspondances
    entre rôles thématiques et rôles sémantiques
    Coder ces correspondances dans T
  fin si
sinon
  Remplacer les rôles thématiques par les rôles sé-
  mantiques codées pour cette structure argumen-
  tale dans T
fin sinon
fin pour

```

FIGURE 4.4 – Algorithme pour faire correspondre les rôles thématiques et les rôles sémantiques

Un programme pour récupérer ces prépositions a également été réalisé, qui énumère pour chaque verbe les différentes prépositions qui ont été trouvées dans le texte. Les résultats renvoyés par les automates de récupération des prépositions doivent être filtrés à la main ; dû au fait que les prépositions sont extraites en corpus, les résultats comportent des erreurs (cf §5.1). Nous avons mis en place un protocole de récupération des prépositions décrit dans la figure 4.5.

1. Annotation des unités auxquelles peuvent être attribué le rôle sémantique d'un argument à extraire.
2. Passage des automates de récupération des prépositions.
3. Classement des prépositions par verbe pour une meilleure visualisation par l'utilisateur
4. Filtrage des aberrations par l'utilisateur
5. Ajout des prépositions sélectionnés à la table de données

FIGURE 4.5 – Méthode de récupération des prépositions

## 4.2 Automates d'extraction

Afin d'extraire les informations sur les rachats d'entreprises, nous avons mis en place une méthode à base d'automates. Cette partie consiste en la description des automates créés, de la méthode avec laquelle ils ont été créés, ainsi que les sorties qu'ils produisent.

### 4.2.1 Format de sortie des automates

Les sorties des automates sont au format XML. Chaque phrase correspondant à une relation à extraire est analysée pour faire ressortir les acteurs de la relation, mais aussi éventuellement des compléments circonstanciels, des adverbes...

Chaque morceau de phrase à extraire est entouré des balises <UNIT> et </UNIT>. Le verbe introduisant la relation est entouré des balises <VERBE>. Les différents arguments sont entourés de balises qui correspondent à leur position dans la grille thématique de Volem. L'argument 1 aura pour balise ARG1, l'argument 2 : ARG2, et le 3 : ARG3. A chacun de ces arguments correspond un rôle sémantique. Ceux-ci sont entourés des balises TH1, TH2 et TH3. Les adverbes sont entourés de balises <ADV>, et les compléments circonstanciels, de balises les décrivant : <CCMONTANT> pour un complément circonstanciel de montant, <CCDATE> pour un complément circonstanciel de date...

Si éventuellement, le verbe est introduit par un verbe « introducteur », comme dans la phrase : « Bull a annoncé avoir racheté CP8 à Schlumberger pour 350 millions d'euros. », ce verbe « introducteur » est entouré de balises SUPP. A noter qu'aucune distinction n'est encore faite entre verbes « introducteurs » et verbes « support ». Dans les phrases :

- Bull **a annoncé** le rachat de CP8 à Schlumberger pour 350 millions d'euros.
- Bull **a procédé** au rachat de CP8 à Schlumberger pour 350 millions d'euros.

Les verbes « annoncer » et les verbes « procéder » sont traités de la même manière, et entourés des même balises SUPP. La différenciation entre verbes support et verbes introducteurs peut se réaliser en aval avec un filtrage par lexique.

Dans le cas où le verbe serait marqué par un conditionnel, une balise MOD entoure le marqueur de ce conditionnel (cela peut être le verbe introducteur autant que le verbe porteur de la relation).

Les adverbes sont entourés de balises ADV. Les adverbes devraient sûrement nécessiter une étude en aval, car ils peuvent être porteurs d'une modalité, d'un conditionnel...

Au sein d'une balise ARG<sub>n</sub>, on peut également constater la présence d'une balise <NOYAU>. Celle-ci correspond à l'entité central dans un groupe correspondant à un argument. Par exemple, le groupe « RCLN, une équipe du LIPN... » serait analysée comme suit : <ARG<sub>n</sub>> <NOYAU> RCLN </NOYAU>, une équipe du LIPN</ARG<sub>n</sub>>.

La phrase « Bull a racheté CP8 à Schlumberger pour 350 millions d'euros » serait analysée par nos automates comme décrit en figure 4.6

Entrée : **Bull a racheté CP8 à Schlumberger pour 350 millions d'euros.**  
 Sortie : <UNIT> <VERBE> racheter </VERBE> <ARG1> <NOYAU> Bull </NOYAU> </ARG1> <TH1>Acheteur</TH1> <ARG2> <NOYAU> CP8 </NOYAU> </ARG2> <TH2>Obj\_Achat</TH2> <ARG3> <NOYAU> Schlumberger </NOYAU> </ARG3> <TH3> Vendeur </TH3> <CCMONTANT> 350 millions d'euros</CCMONTANT>

FIGURE 4.6 – Exemple de sortie au format des automates

## 4.2.2 Automates générateurs

### Méthode de création d'automates et exemples

Unitex permet la construction d'automates générateurs. Ce sont des automates qui, à partir d'une base de données sous une forme semblable aux tables du lexique-grammaire (cf §2.3.2), permet de générer, pour chaque verbe de cette table, un automate qui reconnaîtra une certaine forme syntaxique. Pour créer les automates générateurs, nous avons procédé comme suit :

1. Réflexion sur les formes de surface liées aux alternances de Volem.
2. Création pour chaque alternance de Volem, d'un automate codant les différentes formes de surface qui lui sont liées.

3. Ajout à ces automates générateurs des insertions permettant de gérer les adverbes, les compléments circonstanciels, d'éliminer des propositions subordonnées relatives...
4. Ajout à ces automates des sorties permettant d'obtenir un template au format XML.

Pour comprendre le système d'automates en place, un automate générateur est présenté en figure 4.10. Cet automate simplifié génère les automates reconnaissant les alternances *caus\_2np* (deux groupes nominaux dans la structure du verbe : un sujet et un complément d'objet direct). L'automate original a été amputé d'une grosse partie, et ne reconnaît que les verbes à des temps composés de l'indicatif.

Le premier élément de l'automate, @J, restreint la génération d'automates aux verbes dont la colonne J (celle codant l'alternance *caus\_2np*) contient « + ».

L'élément suivant, @AG, fait référence à la colonne AG de notre table de données, qui correspond au rôle sémantique du premier argument. Dans le graphe généré, cet élément sera donc remplacé par le rôle sémantique joué par le premier argument.

Les deux éléments @B à @C fonctionnent de la même manière que le premier élément décrit (@J) : soit la colonne B (codant l'auxiliaire être) contient un « + », et dans ce cas le lien vers l'élément suivant est activé, soit la colonne C (codant l'auxiliaire avoir) contient un « - », et dans ce cas le lien vers être est désactivé, et celui vers avoir est activé.

L'élément <@A.V :K> récupère la première colonne de la table de données (le verbe à l'infinitif), et est remplacé dans le graphe généré par : « Verbe\_à\_1\_infinitif.V :K ». « .V :K » restreint l'élément aux formes du verbe au participe passé.

L'élément @AH correspond au deuxième argument, et fonctionne de la même manière que @AG.

Un des graphes résultant de la génération de graphes à partir de ce graphe simplifié et de notre table de données est montré en figure 4.11.

### Précisions sur la méthode de création des automates

Les graphes générateurs que nous avons créés n'ont pas été créés à partir d'une approche en corpus. En effet, pour que la méthode que nous avons mise en place puisse rester la plus générale possible afin qu'elle puisse être adaptée à diverses tâches d'extraction, nous avons préféré fonder la génération des automates sur une approche générale de la langue plutôt que sur une approche spécifique à un langage utilisé dans un corpus spécialisé.

Les formes syntaxiques de surface gérées par les automates sont issues d'une formalisation des alternances de Volem. La figure 4.7 décrit le passage d'une alternance de Volem à des formes syntaxiques plus spécifiques.

Les différents automates construits sont détaillés en A.

– **caus\_2np\_pp** :

[ np ] Verbe [ pp ] [ np ] : *Pierre a acheté au bouquiniste un magazine informatique.*

[ np ] Verbe [ np ] [ pp ] : *Pierre a acheté un magazine informatique au bouquiniste.*

A ces deux formes, s'ajoutent les formes avec des insertions : compléments circonstanciels, adverbes...

– **pas\_etre\_part\_np\_2pp** (forme passive) :

[ np ] Verbe [ pp1 ] [ pp2 ] : *Le livre a été racheté par Pierre à Jean.*

[ np ] Verbe [ pp2 ] [ pp1 ] : *Le livre a été racheté à Jean par Pierre.*

Verbe sous forme nominale (proche de la forme passive) : [Nom\_Verbe] [ pp1 ] [ pp2 ] [ pp3 ] : *Rachat à Schlumberger de CP8 par Bull*

FIGURE 4.7 – Exemples du passage d'alternances de Volem à des formes de surface

#### 4.2.3 Automates de récupération des prépositions et programmes liés

Nous avons réalisé deux types d'automates générateurs différents. Le premier type d'automates part d'une ressource complète, comprenant les informations avec lesquelles nous avons enrichi Volem ; l'autre type d'automates part d'une ressource enrichie, mais sans les prépositions qui introduisent généralement les arguments indirects. Ces automates reconnaissent

les mêmes formes syntaxiques que les automates du premier type. Dans ces automates, les prépositions introduisant les arguments codées dans les automates d'extraction du premier type, sont remplacées par un graphe reconnaissant soit une préposition, soit des mots suivis par une préposition, sensé reconnaître des prépositions composées qui ne sont pas présentes dans les dictionnaires d'Unitex (Dela, Delacf...).

Les éléments de texte reconnus comme des prépositions introductrices d'un argument sont stockés dans la structure XML décrite au §4.2.1, entourés par des balises  $PREP_n$ .

Une méthode permettant de compter et classer les prépositions a été mise en place, décrite dans la figure 4.8, et implémentée partiellement. La partie concernant la normalisation des prépositions n'a pas été codée en raison du manque d'informations flexionnelles sur les prépositions (*cf B*)

1. Passage sur un corpus d'entraînement des automates de récupération des prépositions.
2. Normalisation des prépositions présentes dans le fichier XML de sortie.(non implémenté)
3. Pour chaque verbe participant à une alternance reconnue et extraite par les automates, comptage des différentes prépositions.
4. Affichage à l'utilisateur des verbes, des prépositions introduisant leurs arguments et du nombre de ces prépositions.
5. Filtrage par l'utilisateur de ces prépositions, et ajout des prépositions filtrées à la table de données.

FIGURE 4.8 – Méthode pour enrichir Volem avec les prépositions

### 4.3 Filtrage probabiliste des alternances

Il existe des méthodes d'acquisition en corpus de schémas de sous-catégorisation [BC97], [SSA]. Celles-ci partent d'une approche probabiliste des résultats d'analyse syntaxique de corpus. Partant de ce principe, et afin d'améliorer les performances des automates, nous avons inclus aux format de sortie de ceux-ci des balises ALT qui contiennent le nom de l'alternance (codage Volem) qu'une phrase extraite exhibe ; ceci dans le but de fournir à l'utilisateur les outils permettant de filtrer les alternances n'apparaissant

que rarement ou pas du tout, et pourtant définies dans Volem afin d'éliminer les automates qui ne servent pas pour une tâche d'extraction donnée.

Un outil a été également réalisé permettant de référencer pour chacun des verbes d'une table, les alternances qui ont été repérées lors de la tâche d'extraction et leur nombre (*cf C*).

Il serait sûrement utile de réviser la façon dont les graphes sont créés, afin d'avoir non pas un ou deux graphes par alternances Volem, mais un graphe par schéma de sous-catégorisation. Cela permettrait un filtrage plus efficace des alternances, mais nécessiterait de coder d'une manière différente la table de données dans son entier, ainsi que de construire à nouveau tous les graphes créés.

#### 4.4 Ajout d'arguments aux informations à extraire

Les informations contenues dans la table de données créées à partir des données Volem enrichies comme décrit en §4.1.1 ne permettent pas de définir, pour une tâche d'extraction précise, un cadre sémantique exhaustif définissant les informations à extraire pour les relations sur lesquelles porte la tâche d'extraction.

Par exemple, nous avons vu en §2.2.3 que FrameNet définit, dans le cadre cognitif d'une action de vente, un acheteur, un vendeur, l'objet de la transaction, une monnaie d'échange...

Avec la table de données telle que nous l'avons construite, les arguments que nous extrairions dans le cadre d'une extraction sur les rachats d'entreprise seraient : un acheteur, un vendeur, et un objet acheté. L'idéal serait de pouvoir extraire également des arguments qui viendraient compléter ce cadre, afin d'obtenir un cadre semblable à celui décrit dans FrameNet (*commercial\_transaction*).

Considérant une approche semblable à celle de Briscoe[BC97], il est possible de repérer les arguments essentiels d'un sens de verbe en fonction de leur fréquence d'apparition en corpus. Nous avons donc pensé à une méthode, décrite dans la figure 4.9, qui permet de compléter les informations extraites grâce au contenu de notre table de données.

1. Extraction et annotation des phrases à extraire à l'aide des automates décrits en §4.2
2. Pour l'ensemble des phrases extraites, référencement et comptage des adjonctions (compléments circonstanciels) (cf §D)
3. Définition d'un seuil au-delà duquel les adjonctions seront considérés comme faisant partie du cadre de l'extraction.
4. Extraction des adjonctions d'un type dont le seuil est supérieur au seuil défini en 4.

FIGURE 4.9 – Algorithme pour l'ajout d'adjonctions aux arguments à extraire

## 4.5 L'annotation des arguments

L'annotation des unités lexicales pouvant correspondre à un argument d'un des verbes de la table de données a été réalisée de manière ad-hoc. Il devrait être possible de faire le même travail de manière générique, en repérant les groupes de mots faisant intervenir une entité nommée. Par exemple, avec l'entité Bull, on pourrait annoter comme arguments :

- Bull
- Le président de Bull
- 50% du capital de Bull
- la filiale de Bull
- ...

Après avoir constaté que les groupes à annoter dans les corpus spécialisés sur lesquels nous avons travaillé (FirstInvest, FUSACQ-cf 4.6) étaient présentés de la même manière, nous avons décidé d'annoter les arguments d'une manière plus aisée pour nous, mais pas forcément plus performante.

Après avoir repéré les entités nommées en corpus, nous avons créé des automates permettant de repérer des groupes tels que :

- La branche <activité> de BULL
- Le groupe BNP ParisBas
- L'équipementier Nike
- <société>, la filiale française de <société>
- ...

Les résultats obtenus par cette méthode sont mitigées, comme montré en §5.3.

## 4.6 La constitution de corpus

Nous avons fait tourner les automates sur deux types de corpus différents : des corpus spécialisés, et des corpus plus généraux tirés de journaux francophones (*description de la taille des corpus dans la figure 4.12*).

Le plus gros des corpus spécialisés que nous avons utilisé est celui qu'avait déjà utilisé Thierry Poibeau pour l'extraction d'informations sur les rachats d'entreprise[Poi03] (corpus1FirstInvest). Il s'agit d'un corpus de FirstInvest, site spécialisé dans l'économie de marché (First Invest a depuis été renommé Capital) : <http://www.capital.fr>.

Nous avons également constitué un corpus sur les rachats d'entreprise en fusionnant des dépêches tirées du site FUSACQ (fusion/acquisitions) : <http://www.fusacq.com>.

Le dernier corpus spécialisé que nous avons constitué est un corpus tiré de First Invest, mais seulement sur les années 2005-2006.

Un dernier type de corpus a été utilisé : en exécutant des requêtes pour récupérer des phrases contenant des verbes témoins de rachats d'entreprises sur le site <http://glossa.fltr.ucl.ac.be/indexf.html> (site qui passe automatiquement des expressions régulières sur différents journaux), nous avons récupéré et fusionné les phrases correspondant effectivement à des rachats d'entreprises afin de constituer un corpus qui ne soit pas un corpus spécialisé.

Un corpus a été également utilisé pour créer des automates capables de reconnaître les différents groupes qui possèdent les caractéristiques sémantiques d'arguments à extraire. Ce corpus est une partie du premier corpus de FirstInvest.

## 4.7 Conclusion des réalisations

Grâce aux algorithmes proposés et à leur implémentation, nous avons pu coder les différentes informations sur les verbes qu'il manquait à Vo-

lem pour réaliser la tâche d'extraction sur les rachats d'entreprises. Les automates d'extraction ont également été mis en place, et nous allons aborder dans le chapitre suivant l'évaluation de nos différentes méthodes pour compléter les ressources, des résultats obtenus ainsi qu'une deuxième évaluation à posteriori de la ressource Volem afin d'expliquer les résultats de l'extraction d'informations.

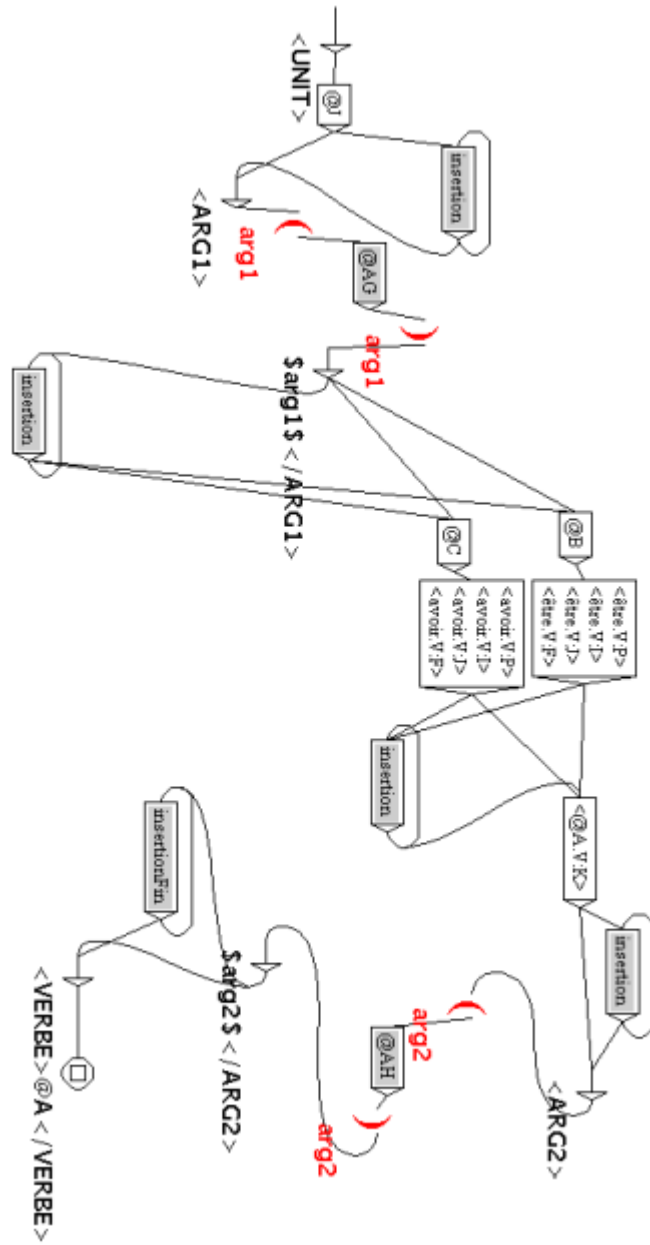


FIGURE 4.10 – Exemple d'un automate générateur

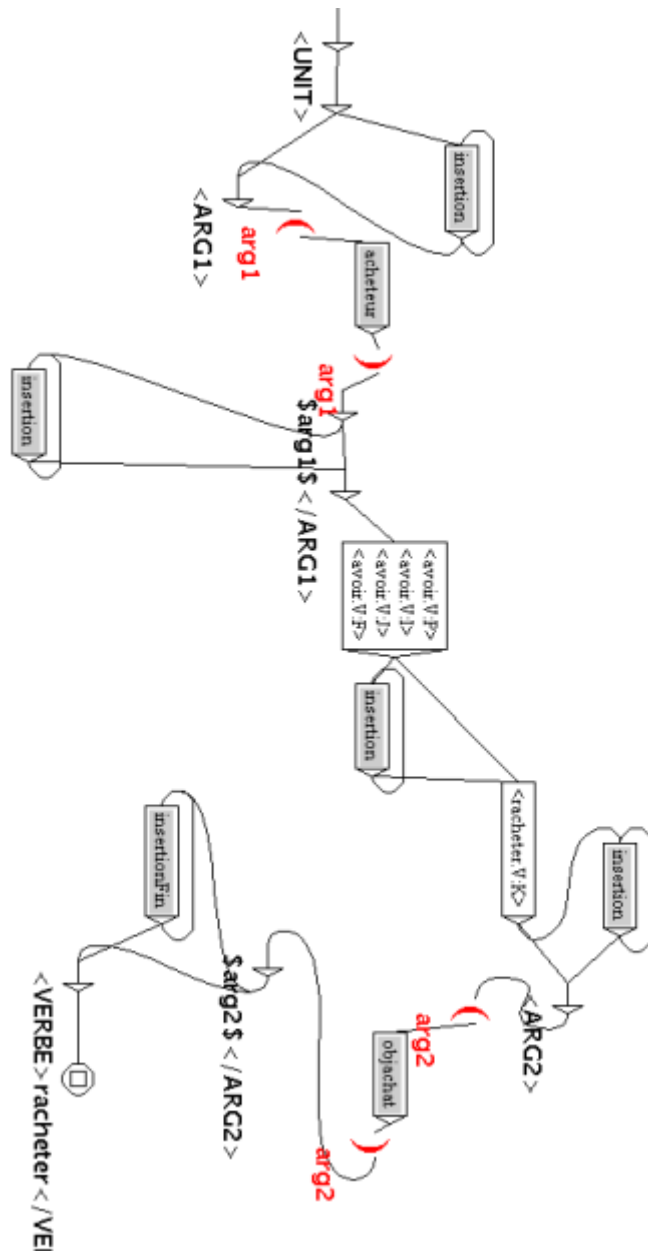


FIGURE 4.11 – Exemple d'un graphe généré

- corpus1FirstInvest : Corpus spécialisé, 717.7ko
- corpus2FirstInvest : Corpus spécialisé, 240ko
- corpusFUSACA : Corpus spécialisé, 298.6 Ko
- corpusGeneral : Corpus tiré de différents journaux non-spécialisés, 390ko
- corpusEntrainement : Corpus spécialisé, 180ko

FIGURE 4.12 – Brève description des différents corpus utilisés pour l'extraction

## Chapitre 5

# Evaluations de l'Extraction et des Ressources

Nous présenterons dans ce chapitre l'évaluation de l'annotation sémantique, l'évaluation des différentes méthodes mises en place pour compléter Volem, l'évaluation de l'extraction d'informations, ainsi que l'évaluation de la ressource utilisée pour réaliser l'extraction d'informations, à savoir Volem enrichie comme décrit en §4.1.1.

### 5.1 Evaluation de l'extraction des prépositions introduisant les arguments

Toutes les prépositions qui introduisent les arguments dans les corpus sont retrouvées. Le problème de la méthode réside dans les erreurs renvoyées, comme « de Gerzat à », ou « Group détenus via ». Certains groupes de mots composés au plus de deux mots suivis d'une préposition simple connue sont reconnus comme des prépositions si le groupe de mots situé derrière est reconnu comme un argument. Cette méthode possède tout de même l'avantage de permettre de coder des prépositions comme « auprès de », auxquelles nous n'aurions pas forcément pensé. Cette méthode nécessite, en raison du taux d'erreurs (25%), un filtrage à la main des groupes de mots reconnus comme des prépositions.

## 5.2 Evaluation de l'extraction en corpus de l'auxiliaire d'un verbe

Nous avons passé l'automate d'extraction des auxiliaires sur le plus gros corpus de FirstInvest. La seule erreur concerne le verbe « détenir » pour lequel seules des formes passives apparaissent. En corpus, nous ne sommes jamais sûr des alternances qui vont être utilisées pour un verbe donné (cf 5.3.4) ; c'est pourquoi il serait préférable d'acquérir les auxiliaires d'un verbe depuis une autre ressource, qui serait complète (dictionnaire électronique ?).

## 5.3 Evaluation de l'extraction d'informations

### 5.3.1 Protocole d'évaluation

Afin d'évaluer l'extraction d'informations réalisée, nous avons mis en place un protocole d'évaluation qui nous permet d'identifier précisément les différentes raisons des résultats obtenus. Le protocole d'évaluation doit nous permettre de nous rendre compte de ces différents points :

- Si l'annotation sémantique semi-automatique donne de bons résultats.
- La pertinence et le caractère complet des résultats en partant d'une annotation sémantique correcte.
- Pourquoi certaines relations n'auraient pas été extraites.

Nous avons donc mis en place un protocole décrit dans la figure 5.1 permettant d'évaluer l'extraction en isolant les différentes étapes de l'extraction qui peuvent être problématiques.

### 5.3.2 Résultats de l'évaluation selon le protocole établi

Le protocole que nous avons décrit dans la figure 5.1 étant extrêmement coûteux en temps (étapes 2 et 3), nous avons choisi de ne l'appliquer qu'à un seul corpus : le corpus de FUSACQ qui est le plus gros des corpus spécialisés que nous avons construit durant ce stage. Les résultats chiffrés obtenus sont décrits dans le tableau 5.1.

1. Passage des automates d'extraction sur le corpus brut.
2. Filtrage du corpus : élimination de toutes les phrases qui ne contiennent pas une relation de rachat d'entreprise.
3. Annotation des différents arguments ainsi que des compléments circonstanciels.
4. Passage des automates d'extraction sur le corpus modifié.
5. Comparaison entre les résultats renvoyés par les automates passés sur le corpus brut et ceux passés sur le corpus modifiés afin d'évaluer l'annotation des arguments.
6. Comparaison des résultats renvoyés par les automates passés sur le corpus modifié et ce même corpus modifié afin d'évaluer la reconnaissance des différentes formes syntaxiques.

FIGURE 5.1 – Protocole d'évaluation de l'extraction d'informations

	Corpus brut	Corpus modifié	Corpus entièrement annoté
Nombre de relations repérées	101	184	286
% de relations	35	64	100

TABLE 5.1 – Tableau de résultats chiffrés de l'extraction

La première constatation est que seulement la moitié des groupes pouvant correspondre à une société, ou un dirigeant sont repérés. Ceci s'explique notamment par le fait que la création manuelle de règles permettant de les repérer a été réalisée en partant d'un corpus d'entraînement extrait d'un corpus issu d'une source différente. De plus, ceci n'étant pas la partie la plus importante de ce mémoire, aucune retouche n'a été faite qui aurait pu améliorer ces résultats.

Un deuxième constat s'impose : malgré l'annotation manuelle des arguments, seules 60% des relations de rachats d'entreprises ont été repérées. Ceci s'explique par le non-codage de certaines formes syntaxiques que nous allons énumérer. Ces formes avaient été énumérées lors de la formalisation des alternances de Volem, mais n'ont pas été codées par la suite en raison notamment de leur ambiguïté sémantique qu'elles peuvent entraîner. Ces

formes sont énumérées dans la section suivante.

### 5.3.3 Les formes syntaxiques existantes en corpus et non reconnues

Les formes syntaxiques non reconnues par les automates que nous avons créés sont de deux types. Soit leur non-reconnaissance est due au fait que, partant des alternances de Volem, les automates créés ne gèrent pas certaines formes syntaxiques ; soit les alternances de Volem ne sont pas assez complètes pour certains verbes, et les automates créés ne gèrent pas des alternances qu'ils devraient repérer.

Voici la liste des formes syntaxiques pouvant être dérivées d'alternances de Volem qui ne sont pas reconnues :

- les subordonnées relatives : *Bull, qui a racheté CP8 à Schlumberger...*
- les verbes introducteurs ou support précédés d'un verbe marquant soit le futur, soit le passé : *Bull vient d'annoncer le rachat de CP8 à Schlumberger*
- les structures complexes telles que : *COMPANY s'était diversifié à travers l'acquisition de COMPANY, ou COMPANY a manifesté son intérêt pour une éventuelle reprise de COMPANY*
- Les propositions participiales : *Ayant racheté COMPANY, COMPANY...*
- D'autres constructions pouvant être dérivées d'une alternance Volem comme : *COMPANY fait un pas vers le « C to C » en rachetant la COMPANY.*

Voici maintenant l'alternance qui n'est pas présente dans Volem :

- l'alternance `pas_etre_part_np` : *COMPANY, acquis CCMONTANT, ...*

### 5.3.4 Un filtrage des alternances inutiles est-il possible ?

Afin de filtrer les alternances qui n'apparaissent pas en corpus, comme décrit en §4.3, il est nécessaire de savoir quelles alternances sont codées qui n'ont pas été repérées dans le corpus qui sert à l'extraction. Nous avons décidé d'évaluer la pertinence de cette méthode en comparant les résultats obtenus dans chacun des corpus que nous avons utilisés. Cette analyse nous permettra de nous rendre compte si cette méthode peut s'appliquer à un ensemble de corpus supports d'une tâche d'extraction donnée, ou si

Verbe	Alternances	nombre d'occurrences de l'alternance
racheter	caus_2np	32
	caus_2np_pp	1
	pas_etre_part_np_pp	1
revendre	caus_2np	1
acheter	caus_2np	2
vendre	caus_2np	8
	caus_2np_pp	5
acquérir	caus_2np	26
	caus_2np_pp	1
céder	caus_2np	19
fusionner	caus_2np_pp	3
	caus_np_plu	2
	caus_2np	2
détenir	caus_2np	18
	pas_etre_part_np_pp	5
offrir	caus_2np_pp	1
	caus_2np	1

TABLE 5.2 – Répartition des alternances selon les verbes dans les phrases extraites des corpus (FirstInvest)

chaque corpus, même spécialisé, possède ses propres spécificités d'écriture, ce qui rendrait la méthode moins applicable. Les résultats obtenus sur les différents corpus sont décrits dans les tableaux 5.2, 5.3, 5.4

On peut constater des différences importantes en ce qui concerne les alternances utilisées dans les deux corpus spécialisés. Des phrases du type « COMPANY s'achète COMPANY pour MONTANT » sont présentes dans le corpus de FUSACQ, mais pas dans celui de FirstInvest (alternance caus\_refl\_

pr\_np). Quant au corpus tiré de journaux plus généralistes, il n'est pas réellement possible de tirer de conclusions en l'état actuel des choses : les journaux utilisent un style plus littéraire pour réécrire les dépêches qu'ils reçoivent. Ainsi, plus de formes subjonctives et participiales sont présentes qui ne sont pas reconnues. De même, dans ces corpus, les relations de rachats d'entreprises sont décrites au sein de récits descriptifs ; les noms d'entreprises et de dirigeants sont donc souvent généralement remplacés par des pronoms, qui ne sont pas reconnus par nos graphes de reconnaissance de syntagmes typés sémantiquement comme « Acheteur », « Vendeur » ou « Objet Acheté ».

Verbe	Alternances	nombre d'occurrences de l'alternance
racheter	caus_2np	37
	caus_2np_pp	6
	pas_etre_part_np_pp	6
revendre	caus_2np	1
acheter	caus_refl_pr_np	2
vendre	pas_etre_part_np_2pp	2
	caus_2np	2
	caus_2np_pp	2
acquérir	caus_2np	44
	caus_2np_pp	1
céder	caus_2np_pp	24
	caus_2np	9
	pas_etre_part_np_2pp	2
fusionner	aucune occurrence	0
détenir	caus_2np	5
	pas_etre_part_np_pp	1
offrir	caus_2np_pp	1
	caus_2np	1
reprendre	pas_etre_part_np_pp	11
	caus_2np	12

TABLE 5.3 – Répartition des alternances selon les verbes dans les phrases extraites des corpus (FUSACQ annoté)

Verbe	Alternances	nombre d'occurrences de l'alternance
racheter	caus_2np	8
	caus_2np_pp	6
	pas_etre_part_np_pp	6
revendre	aucune occurrence	0
acheter	caus_2np	4
vendre	aucune occurrence	0
acquérir	pas_etre_part_np_pp	4
	caus_2np_pp	1
céder	caus_2np_pp	1
	pas_etre_part_np_2pp	2
fusionner	aucune occurrence	0
détenir	pas_etre_part_np_pp	3
offrir	caus_2np_pp	4
reprendre	pas_etre_part_np_pp	4
	caus_2np	2

TABLE 5.4 – Répartition des alternances selon les verbes dans les phrases extraites des corpus (Corpus général)

Le filtrage des alternances par des méthodes probabilistes ne sont donc valables que pour un corpus, dans le cadre par exemple d'un programme de veille. En effet, il se peut, en changeant de corpus, que le style d'écriture ne soit pas le même ; ce qui aura pour effet de changer les alternances qui interviennent dans le texte.



## Chapitre 6

# Conclusions et perspectives

Après avoir évalué la ressource que nous avons créée, nous sommes en mesure de tirer des conclusions sur la pertinence de cette ressource pour une tâche d'extraction, et d'établir des comparaisons entre la ressource que nous avons obtenue et les ressources anglaises.

### 6.1 Conclusions

L'état de l'art montre une réelle différence au niveau de l'avancement des ressources pour l'anglais et pour le français, ce qui peut présenter un handicap pour le traitement automatique de la langue française.

L'applicabilité de la ressource que nous avons créée à partir de données Volem et d'autres acquises en corpus est réelle : cette ressource nous permet d'extraire, si l'annotation sémantique des groupes de mots qui correspondent à d'éventuelles entités à extraire est correcte, 65% des relations sur les rachats d'entreprises. Sur les 35% restants, environ 25% sont dus au non-codage de certaines règles syntaxiques acquises à partir de cette ressource, et 5% au non-codage dans la ressource elle-même de certaines règles.

Un travail a été mené par Ana-Maria Giuglea et Alessandro Moschitti [AMG04], qui permet, à partir des trois ressources lexicales pour l'anglais FrameNet, VerbNet, et PropBank, d'annoter sémantiquement, d'après les éléments de cadre de FrameNet, les arguments des verbes en corpus. L'outil qu'ils ont créé permet d'annoter sémantiquement et correctement 91% des

arguments.

On peut supposer qu'en disposant des mêmes ressources pour le français, il serait possible d'obtenir les mêmes résultats, et ce de manière totalement automatique (pas d'annotation préalable des syntagmes qui correspondent à des arguments à extraire).

La création d'une ressource complète pour le français peut donc apparaître comme une étape nécessaire pour améliorer le traitement automatique des langues. Les ressources dont on dispose actuellement pour le français ne sont en effet pas suffisantes, tant au niveau de la description des entrées que de la quantité de celles-ci.

## 6.2 Perspectives

Les automates que nous avons générés ne gèrent pas l'ensemble des alternances de Volem. Une perspective pourrait être de créer les automates qui gèrent les alternances restantes. On pourrait également tenter une approche pour l'annotation des arguments éventuels fondée sur la reconnaissance des groupes syntaxiques comportant certaines propriétés (comme le fait qu'ils contiennent une entité nommée du type que l'on souhaite extraire), et utiliser des outils de résolutions d'anaphores à lier avec les automates déjà créés afin d'éviter la multiplication d'automates pour la reconnaissance de propositions subjonctives.

# Bibliographie

- [AMG04] Alessandro Moschitti Ana-Maria Giuglea. Knowledge discovering using framenet, verbnet and propbank. In *International Workshop on Mining for and from the Semantic Web*, Seattle, USA, 2004.
- [BC97] T. Briscoe and J. Carroll. Automatic extraction of subcategorization from corpora, 1997.
- [BFL98] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The Berkeley FrameNet project. In Christian Boitet and Pete White-lock, editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 86–90, San Francisco, California, 1998. Morgan Kaufmann Publishers.
- [CJ01] Collin F. Baker Charles J.Fillmore. Frame semantics for text understanding. In *WordNet and Other Lexical Resources Workshop*, NAACL, Pittsburgh, 2001.
- [EG05] Didier Bourigault Edith Galy. Analyse distributionnelle de corpus de langue générale et synonymie. In *4èmes Journées de la Linguistique de Corpus*, Lorient, France, 2005.
- [FSDV<sup>+</sup>02] Ana Fernandez, Patrick Saint-Dizier, Gloria Vazquez, Mouna Kamel, and Farah Benamara. The Volem Project : a Framework for the Construction of Advanced Multilingual Lexicons . In *Language Technology 2002*, pages 123–142, Hyderabad, décembre 2002. Springer Verlag, Lecture Notes. Dates de conférence : décembre 2002.
- [GGFP05] Claire Gardent, Bruno Guillaume, Ingrid Falk, and Guy Perrier. Le lexique-grammaire de m. gross et le traitement automatique des langues, 2005.
- [GGPF06] Claire Gardent, Bruno Guillaume, Guy Perrier, and Ingrid Falk. Extraction d'information de sous-catégorisation à partir des

- tables du LADL. In *TALN 2006*, Leuven, 2006. 10-13 Avril 2006.
- [Gro86] Maurice Gross. *Grammaire Transformationnelle du Français*. Cantilène, 1986.
- [Kor] Anna Korhonen. Automatic extraction of subcategorization frames from corpora - improving filtering with diathesis alternations.
- [KS05] Karin Kipper Schuler. *VerbNet : A Broad-Coverage, Comprehensive Verb Lexicon*. PhD thesis, Faculties of the University of Pennsylvania, 2005.
- [LBF97] J. Lowe, C. Baker, and C. Fillmore. A frame-semantic approach to semantic annotation, 1997.
- [Lev93] Beth Levin. *English Verb Classes and Alternations : a preliminary investigation*. University of Chicago Press, Chicago and London, 1993.
- [MBF<sup>+</sup>90] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. Introduction to wordnet : An on-line lexical database\*. *Int J Lexicography*, 3(4) :235–244, January 1990.
- [Min] Marvin Minsky. A framework for representing knowledge.
- [PGK05] Martha Palmer, Daniel Gildea, and Paul Kingsbury. The proposition bank : an annotated corpus of semantic roles. *Computational Linguistics*, 31, 2005.
- [Pit06] Guillaume Pitel. Framenet, théorie, produit, processus, multilingualité et connexions. In *Autour de FrameNet et de la Sémantique Lexicale Multilingue : projets en cours et points de contacts entre les différentes approches*, 2006. date de la conférence : 28 Février 2006.
- [Plo] Sabine Ploux.
- [Poi03] Thierry Poibeau. *Extraction Automatique d'Information - du texte brut au web sémantique*. Lavoisier, 2003.
- [SD98] Patrick Saint-Dizier. An introduction to lexical semantics of predicative forms. In Patrick Saint-Dizier, editor, *Cours ESSLI et Predicative Forms in Natural Language and in Predicative Forms*, pages 1–49. Kluwer Academic, Cambridge, USA, juillet 1998.
- [SSA] Paula Chesley Susanne Salmon-Alt. Le filtrage probabiliste dans l'extraction automatique de cadres de sous-catégorisation.

## Annexe A

# Présentation des différents automates créés pour l'extraction d'informations

**anti\_pr\_np.grf** reconnaît les phrases du type : « ARG pronom Verbe »

**anti\_pr\_np\_pp.grf** reconnaît les phrases du type : « ARG ponom Verbe Arg<sub>pp</sub> »

**caus\_2np.grf** reconnaît les phrases du type : « ARG Verbe ARG »

**caus\_2np\_pp.grf** reconnaît les phrases du type : « ARG Verbe ARG ARG<sub>pp</sub> »

**caus\_npplu.grf** reconnaît les phrases du type : « ARG, ARG et ARG Verbe »

**caus\_refl\_pr\_2np** reconnaît les phrases du type : « ARG pronom réflexif Verbe ARG »

**nom\_pas\_etre\_part\_np\_2pp.grf** reconnaît les phrases du type : « Verbe (forme nominale) de ARG par ARG ARG<sub>pp</sub> »

**pas\_etre\_part\_np\_2pp.grf** reconnaît les phrases du type : « ARG Verbe (passif) par ARG ARG<sub>pp</sub> »

**pas\_etre\_part\_np\_pp.grf** reconnaît les phrases du type : « ARG Verbe (passif) par ARG »

**support\_caus\_2np\_pp.grf** reconnaît les phrases du type : « ARG Verbe\_ support Verbe (sous forme nominale) ARG ARG<sub>pp</sub> »

**support\_caus\_2np.grf** reconnaît les phrases du type : « ARG Verbe\_ support Verbe (sous forme nominale) ARG »

**verbesupport\_caus\_2np\_pp.grf** reconnaît les phrases du type : « ARG Verbe\_ support Verbe ARG ARG<sub>pp</sub> »

`verbesupport_caus_2np.grf`] reconnaît les phrases du type : « ARG Verbe\_-support Verbe ARG »

**`verbesupport_pas_etre_part_np_np_2pp.grf`** reconnaît les phrases du type :  
« ARG Verbe\_-  
support par ARG ARG<sub>pp</sub> »

**`verbesupport_pas_etre_part_np_pp.grf`** reconnaît les phrases du type :  
« ARG Verbe\_support par ARG »

Ces automates ont été également refaits en partie pour travailler à l'extraction des prépositions.

## Annexe B

# Programme pour les statistiques sur les prépositions

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define NBVERBES 100
#define NBPREP 300

typedef struct {
    unsigned int nbOcc;
    char *nomPrep;
} comptePrep;

typedef struct {
    char *nomVerbe;
    unsigned int nbOcc;
    comptePrep prepositions[NBPREP];
    int nbPrepositions;
} verbe;

typedef struct {
    verbe structVerbes[NBVERBES];
    int nbVerbes;
} verbes;

void inscrirePrep ( char *p, char *nv, verbes *sv);
void visualiserPreps ( verbes *sv );
int inscrirePrepDansVerbe ( char *p, verbe *v );

int main (int argc, char **argv)
{
    verbes listeVerbesPrep;

    char c;
    char nomPrep [100];
    char nomBalise [6];
    int numeroPrep, i, testVerbe;
    char nomVerbe [50];

    FILE *fIn;

    if (argc != 2)
    {
        printf ("Syntaxe: %s [ fichier_courant ]\n", argv [0]);
        return (0);
    }
}
```

```

if ( ( fIn = fopen (argv[1], "r") ) == NULL)
{
    printf ("Impossible_d'ouvrir_le_fichier_%s_en_lecture\n", argv[1]);
    return (-1);
}

//initialisation listeVerbesPrep
listeVerbesPrep.nbVerbes = 0;
listeVerbesPrep.structVerbes[0].nbPrepositions = 0;

while ( c != EOF )
{
    c = fgetc (fIn);

    //Parcours le fichier a la recherche d'une balise

    if ( c == '<' )
    {
        //Une balise a ete trouvee. Il faut maintenant savoir s'il s'agit
        //ou non d'une balise <PREP>
        fgets (nomBalise, sizeof(char)*5, fIn);
        //printf("%s", nomBalise);
        if (strcmp (nomBalise, "PREP") == 0)
        {
            //La balise trouvee est bien une PREP. on cherche
            //maintenant le numero de cette PREP
            c = fgetc (fIn);
            numeroPrep = c-48;
            c = fgetc (fIn);

            //printf ("%d\n", numeroPrep);

            //On cherche maintenant l'element que contient
            //cette balise : la preposition en elle-meme
            i = 0;
            nomPrep[i] = fgetc(fIn);
            while ( nomPrep[i] != '<')
            {
                i++;
                nomPrep[i] = fgetc(fIn);
            }
            nomPrep[i] = '\0';

            //Maintenant que l'on a l'element de PREPn,
            //il faut chercher la balise VERBE qui va avec

            c = fgetc (fIn);
            testVerbe = 0;

            while (!testVerbe)
            {
                c = 'a';
                while ( c != '<')
                {
                    c = fgetc (fIn);
                }

                fgets (nomBalise, sizeof(char)*6, fIn);
                //printf("%s\n", nomBalise);

                if ( strcmp (nomBalise, "VERBE") == 0 )
                {
                    testVerbe = 1;
                    c = fgetc (fIn);
                    i = 0;
                    nomVerbe[i] = fgetc (fIn);
                    while (nomVerbe[i] != '<') {
                        i++;
                        nomVerbe[i] = fgetc (fIn);
                    }

                    nomVerbe[i] = '\0';
                }
            }

            printf ("\n\n%s,%s\n", nomPrep, nomVerbe);
            inscrirePrep (nomPrep, nomVerbe, &listeVerbesPrep);

```

```

    }
}

visualiserPreps ( &listeVerbesPrep );

return (1);
}

//Fonction servant a inscrire une preposition p
//dans une structure verbe v.
int inscrirePrepDansVerbe ( char *p, verbe *v ) {
    int i = 0;
    int test = 0;
    //Si le nombre de prepositions n'est pas 0,
    //parcours de la liste des prepositions a la recherche de p
    if (v->nbPrepositions != 0)
        for (i = 0; i < v->nbPrepositions; i++)
        {
            if (strcmp (p,v->prepositions[i].nomPrep) == 0)
            {
                test = 1;
                v->prepositions[i].nbOcc++;
                return(1);
            }
        }

    //Sinon, creation d'une nouvelle preposition
    v->prepositions[v->nbPrepositions].nomPrep =
        (char *) malloc ( sizeof (char) * (strlen (p) + 1));
    strcpy(v->prepositions[v->nbPrepositions].nomPrep,p);
    v->prepositions[v->nbPrepositions].nbOcc = 1;
    v->nbPrepositions++;

    return (1);
}

//Inscrit la preposition p du verbe nv dans le champ qui lui
//est destine de la structures verbes en entree
void inscrirePrep ( char *p,char *nv,verbes *sv ) {
    int i ,test = 0;

    if (sv->nbVerbes != 0)
        for (i =0;i < sv->nbVerbes; i++)
        {
            if (strcmp (sv->structVerbes[i].nomVerbe,nv) == 0)
            {
                test = 1;
                inscrirePrepDansVerbe (p,&(sv->structVerbes[i]));
                sv->structVerbes[i].nbOcc++;
                i = sv->nbVerbes;
            }
        }
    //Si le verbe en question n'existe pas
    if (test == 0)
    {
        sv->structVerbes[sv->nbVerbes].nomVerbe =
            (char *) malloc ( (strlen(nv) + 1) * sizeof (char) );
        sv->structVerbes[sv->nbVerbes].nbPrepositions = 0;
        strcpy (sv->structVerbes[sv->nbVerbes].nomVerbe,nv);
        sv->structVerbes[sv->nbVerbes].nbOcc = 1;
        inscrirePrepDansVerbe (p,&(sv->structVerbes[sv->nbVerbes]));
        sv->nbVerbes++;
    }
}

void visualiserPreps ( verbes *sv ) {
    int i,j;

    for (i = 0; i < sv->nbVerbes; i++)
    {
        printf("\n\n*****_%s_*****\n\n",sv->structVerbes[i].nomVerbe);
        printf("Prepositions :\n");

        for (j = 0; j < sv->structVerbes[i].nbPrepositions; j++)
        {
            printf ("%s : %d occurrences\n",sv->structVerbes[i].prepositions[j].nomPrep,

```

```
sv->structVerbes[i].prepositions[j].nbOcc);  
    }  
}
```

## Annexe C

# Programme pour les statistiques sur les alternances

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define NBVERBES 100
#define NBALT 300

typedef struct{
    unsigned int nbOcc;
    char *nomAlt;
} compteAlt;

typedef struct{
    char *nomVerbe;
    unsigned int nbOcc;
    compteAlt alt[NBALT];
    int nbAlts;
} verbe;

typedef struct{
    verbe structVerbes[NBVERBES];
    int nbVerbes;
} verbes;

void inscrireAlt ( char *a, char *nv, verbes *sv );
void visualiserAlts ( verbes *sv );
int inscrireAltDansVerbe ( char *a, verbe *v );

int main (int argc, char **argv)
{
    verbes listeVerbesAlt;
    int testAlt;
    char c;
    char nomAlt [100];
    char nomBalise [6];
    int numeroAlt, i, testVerbe;
    char nomVerbe [50];

    FILE *fIn;

    if (argc != 2)
    {
        printf ("Syntaxe : %s [ fichier_courant ]\n", argv [0]);
        return (0);
    }
}
```

```

if ( ( fIn = fopen (argv[1], "r") ) == NULL)
{
    printf ("Impossible d'ouvrir le fichier %s en lecture\n", argv [1]);
    return (-1);
}

//initialisation listeVerbesAlt
listeVerbesAlt.nbVerbes = 0;
listeVerbesAlt.structVerbes[0].nbAlts = 0;

while ( c != EOF )
{
    c = fgetc (fIn);
    testVerbe=0;
    //Parcours le fichier a la recherche d'une balise VERBE
    while (!testVerbe && c!= EOF)
    {
        c = 'a';
        while ( c != '<' && c != EOF)
        {
            c = fgetc (fIn);
        }

        if (c != EOF)
        {
            fgets (nomBalise, sizeof(char)*6, fIn);
            //printf("%s\n", nomBalise);

            if ( strcmp (nomBalise, "VERBE") == 0 )
            {
                testVerbe = 1;
                c = fgetc (fIn);
                i = 0;
                nomVerbe[i] = fgetc (fIn);
                while (nomVerbe[i] != '<') {
                    i++;
                    nomVerbe[i] = fgetc (fIn);
                }
                nomVerbe[i] = '\0';
            }
        }
    }

    testAlt = 0;
    if (c!= EOF)
        while(!testAlt)
        {
            c = fgetc (fIn);
            printf("%c", c);
            if ( c == '<' )
            {
                //Une balise a ete trouvee. Il faut maintenant
                //savoir s'il s'agit ou non d'une balise <ALT>
                fgets (nomBalise, sizeof(char)*4, fIn);
                //printf("%s", nomBalise);
                if (strcmp (nomBalise, "ALT") == 0)
                {
                    testAlt=1;

                    //On cherche maintenant l'element
                    //que contient cette balise :
                    //l'alternance en elle-meme
                    i = 0;
                    nomAlt[i] = fgetc(fIn);
                    while ( nomAlt[i] != '<' )
                    {
                        i++;
                        nomAlt[i] = fgetc(fIn);
                    }
                    nomAlt[i] = '\0';

                    //Maintenant que l'on a l'element
                    //de ALT, il faut chercher la balise
                    //VERBE qui va avec

```

```

        c = fgetc (fIn);
        testVerbe = 0;

        printf("\n\n%s,%s\n", nomAlt,
            nomVerbe);
        inscrireAlt(nomAlt, nomVerbe,
            &listeVerbesAlt);
    }
}

printf("EOF\n");
visualiserAlts ( &listeVerbesAlt );

return (1);
}

//Fonction servant a inscrire une alternance p dans une structure verbe v.
int inscrireAltDansVerbe ( char *a, verbe *v ) {
    int i = 0;
    int test = 0;
    //Si le nombre d'alternances n'est pas 0, parcours de la liste des alternances
    // a la recherche de p
    if (v->nbAlts != 0)
        for (i = 0; i < v->nbAlts; i++)
        {
            if (strcmp (a, v->alt[i].nomAlt) == 0)
            {
                test = 1;
                v->alt[i].nbOcc++;
                return (1);
            }
        }

    //Sinon, creation d'une nouvelle ALTosition
    v->alt[v->nbAlts].nomAlt = (char *) malloc ( sizeof (char) *
        (strlen (a) + 1));
    strcpy(v->alt[v->nbAlts].nomAlt, a);
    v->alt[v->nbAlts].nbOcc = 1;
    v->nbAlts++;

    return (1);
}

//Inscrit l'alternance a du verbe nv dans le champ qui lui est destine
//de la structures verbes en entree
void inscrireAlt ( char *a, char *nv, verbes *sv ) {
    int i, test = 0;

    if (sv->nbVerbes != 0)
        for (i = 0; i < sv->nbVerbes; i++)
        {
            if (strcmp (sv->structVerbes[i].nomVerbe, nv) == 0)
            {
                test = 1;
                inscrireAltDansVerbe (a,
                    &(sv->structVerbes[i]));
                sv->structVerbes[i].nbOcc++;
                i = sv->nbVerbes;
            }
        }

    //Si le verbe en question n'existe pas
    if (test == 0)
    {
        sv->structVerbes[sv->nbVerbes].nomVerbe = (char *) malloc ( (strlen(nv) + 1) *
            sizeof (char) );
        sv->structVerbes[sv->nbVerbes].nbAlts = 0;
        strcpy(sv->structVerbes[sv->nbVerbes].nomVerbe, nv);
        sv->structVerbes[sv->nbVerbes].nbOcc = 1;
        inscrireAltDansVerbe (a, &(sv->structVerbes[sv->nbVerbes]));
        sv->nbVerbes++;
    }
}

void visualiserAlts ( verbes *sv ) {
    int i, j;

```

```
for (i = 0; i < sv->nbVerbes; i++)
{
    printf("\n\n*****%s*****\n\n",
        sv->structVerbes[i].nomVerbe);
    printf("Alternances:\n");

    for (j = 0; j < sv->structVerbes[i].nbAlts; j++)
    {
        printf("%s: occurrences\n",
            sv->structVerbes[i].alt[j].nomAlt,
            sv->structVerbes[i].alt[j].nbOcc);
    }
}
}
```

## Annexe D

# Programme pour les statistiques sur les adjonctions

```
#!/usr/bin/perl -w
use XML::DOM;

my $parser = new XML::DOM::Parser;
my $doc = $parser->parsefile ($ARGV[0]);

#recherche et compte les occurrences de <UNIT>
#(<UNIT> = unite reconnue), de <CCDATE>, de <MONTANT>,
#de <CCLIEU>...
my $unit = $doc->getElementsByTagName ("UNIT");
my $ccdate = $doc->getElementsByTagName ("CCDATE");
my $cclieu = $doc->getElementsByTagName ("CCLIEU");
my $ccmontant = $doc->getElementsByTagName ("MONTANT");

my $nbunit = $unit->getLength;
my $nbccdate = $ccdate->getLength;
my $nbcclieu = $cclieu->getLength;
my $nbccmontant = $ccmontant->getLength;

print "nombre_d' unites_:" . $nbunit . "\n";
print "nombre_d' unites_comportant_ccdate_:" . $nbccdate . "\n";
print "nombre_d' unites_comportant_cclieu_:" . $nbcclieu . "\n";
print "nombre_d' unites_comportant_montant_:" . $nbccmontant . "\n";
```