UNIVERSITÉ DU QUÉBEC À MONTRÉAL

REPRÉSENTATION DE COLLOCATIONS DANS UN RÉSEAU LEXICAL À L'AIDE DES FONCTIONS LEXICALES ET DES FORMALISMES DU WEB SÉMANTIQUE

THÈSE

PRÉSENTÉE

COMME EXIGENCE PARTIELLE

DU DOCTORAT EN INFORMATIQUE COGNITIVE

PAR

ALEXSANDRO FERNANDES DA FONSECA

JUILLET 2018

REMERCIEMENTS

Je remercie :

Mon épouse, Anya, pour son amour inconditionnel, son support et sa patience pendant tout mon parcours au doctorat et la production de la présente thèse ;

Mes parents, Francisco et Adelia, pour avoir cru en moi et être fiers de mes accomplissements ;

Ma directrice de recherche, Fatiha Sadat, pour avoir cru en mon potentiel, pour l'encouragement, pour m'avoir guidé et m'avoir aidé à toujours donner mon meilleur ;

Mon codirecteur de recherche, François Lareau, pour m'avoir aidé à mieux comprendre les concepts de la théorie Sens-Texte ;

Alexandre Blondin Massé, mon premier codirecteur de recherche, pour m'avoir aidé dans l'élaboration de mon projet de thèse ;

Mes professeurs à l'UQAM, en particulier Petko Valtchev, pour m'avoir présenté les concepts du Web sémantique, et Étienne Harnad, pour m'avoir aidé à comprendre plus profondément les concepts des sciences cognitives ;

Alain Polguère, pour m'avoir fourni les données du Réseau lexical du français (RLF), pour m'avoir accepté pour un stage de recherche à l'Université de Lorraine et pour m'avoir expliqué les concepts de la théorie Sens-Texte appliqués au RLF ;

Les membres du jury de ma soutenance de thèse, Alain Polguère, Ophélie Tremblay, Petko Valtchev, François Lareau et Fatiha Sadat, pour la révision de cette

thèse ainsi que leurs corrections et commentaires précieux ;

Mes collègues et amis du laboratoire GDAC, en particulier Tan Le, Fatma Malek et Billal Belainine, pour la compagnie, le support et l'aide technique en plusieurs situations ;

Mes collègues et amis du programme de doctorat, en particulier Ludovic Bocken, Alberto Monteiro et Sasha Luccioni ;

Mon frère, Fernando Fonseca, pour ses encouragements et son support ;

Hortência Nunes, la directrice de mon école primaire à Formiga (Brésil), pour son support et son amitié des 30 dernières années, et pour m'avoir donné les conditions nécessaires pour initier mon parcours académique ;

Tous mes amis qui m'ont supporté, dans le passé ou dans le présent, de proche ou de loin, en particulier : Patrick et Annie Kelly, Eduardo et Ângela Amorim, Sasha et Sveta Obikhod, Ralph et Karina Bernardes, Stanley et Dominique Dumornay, Marc-André Laverdière et Ghislain Normand ;

Finalement, je remercie le FRQNT, la Fondation de l'UQAM et la Faculté des sciences de l'UQAM pour les bourses, les subventions et le support financier général ayant permis la réalisation de la présente recherche.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ADJ	*Adjectif*
API	*Application Programming Interface* (Interface de programmation applicatif)
ART	*Article*
ASem	*Actant Sémantique*
ASyntP	*Actant Syntaxique Profond*
ATILF	*Analyse et Traitement Informatique de la Langue Française*
ATTR	*Attribut*
CE	*Classe d'Équivalence*
CO	*Complément d'Objet*
COdir	*Complément d'Objet direct*
DET	*Déterminant*
DL	*Description Logic* (Logique de description)
DuELME	*Dutch Electronic Lexicon of Multiword Expressions* (Lexique électronique néerlandais d'expressions polylexicales)
EPL	*Expression Polylexicale*
FE	*Frame Element* (Élément de cadre)
FL	*Fonction Lexicale*
FOAF	*Friend Of A Friend* (Ami d'un ami)
FWC	*Free Word Combination* (Combinaison de mots libre)
GC	*Graphe Conceptuel*
GE	*Graphe Existentiel*

HPSG	*Head-driven Phrase Structure Grammar* (Grammaire syntagmatique guidée par les têtes)
ISO	*International Organization for Standardization* (Organisation internationale de normalisation)
KIF	*Knowledge Interchange Format* (Format d'échange de connaissances)
LFG	*Lexical-Functional Grammar* (Grammaire lexicale-fonctionnelle)
LMF	*Lexical Markup Framework* (Cadre de balisage lexical)
MAV	*Matrice Attribut-Valeur*
MCE	*Méthode des Classes d'Équivalence*
N	*Nom*
NP	*Noun Phrase* (Phrase nominale)
OWL	*Web Ontology Language* (Langage ontologique pour le Web)
PRED	*Prédicat*
PREP	*Préposition*
RDF	*Resource Description Framework* (Cadre de description de ressource)
RDFS	*Resource Description Framework schema* (Schéma de cadre de description de ressource)
RELIEF	*REssource Lexicale Informatisée d'Envergure sur le Français*
RL	*Réseau Lexical*
RLF	*Réseau Lexical du Français*
RMorphP	*Réseau Morphologique Profond*
RMorphS	*Réseau Morphologique de Surface*
RPhonP	*Réseau Phonologique Profond*
RPhonS	*Réseau Phonologique de Surface*

RSem	*Réseau Sémantique*
RSyntP	*Réseau Syntaxique Profond*
RSyntS	*Réseau Syntaxique de Surface*
SemEval	*Semantic Evaluation* (Évaluation sémantique)
SG	*Sujet Grammatical*
SKOS	*Simple Knowledge Organization System* (Système simple pour l'organisation des connaissances)
SPARQL	*SPARQL Protocol and RDF Query Language* (Langage de requête RDF et protocole SPARQL)
SQL	*Structured Query Language* (Langage de requête structurée)
SUMO	*Suggested Upper Merged Ontology* (Ontologie fusionnée de haut niveau)
SUO-KIF	*Standard Upper Ontology - Knowledge Interchange Format* (Ontologie de haut niveau standard - Format d'échange de connaissances)
SVM	*Support Vector Machine* (Machines à vecteurs de support)
TAL	*Traitement Automatique des Langues*
TST	*Théorie Sens-Texte*
UNL	*Universal Networking Language* (Langage universel de réseau)
URI	*Uniform Resource Identifier* (Identifiant uniforme de ressource)
VP	*Verbal Phrase* (Phrase verbale)
W3C	*World Wide Web Consortium*
XLE	*Xerox Linguistics Environment* (Environement linguistique Xerox)
XMG	*eXtensible Meta Grammar* (Méta-grammaire extensible)
XML	*eXtensible Markup Language* (Langage de balisage extensible)
YAGO	*Yet Another Great Ontology* (Encore une autre grande ontologie)

RÉSUMÉ

Les collocations posent problème à plusieurs applications liées au domaine du traitement automatique des langues, comme la traduction automatique, la recherche d'information et la génération automatique de texte. Notamment, la représentation des collocations reste un problème ouvert et peu exploité.

Le formalisme des fonctions lexicales a été développé pour représenter les nombreux types de relations possibles entre les mots, comme la synonymie (*voiture - automobile*), l'antonymie (*grand - petit*) et l'hyponymie (*chat - mammifère*) — qui sont des relations paradigmatiques —, et des relations comme l'intensification (*critique virulente*) et la réalisation (*purger une peine*) — qui sont des relations syntagmatiques. La relation entre les constituants d'une collocation est de type syntagmatique et peut être modélisée par les fonctions lexicales du même type.

Dans une relation modélisée par une fonction lexicale, on peut identifier trois parties : la *fonction* (comme synonymie, intensification ou réalisation), la *base* ou *mot-clé*, qui est choisie librement (les mots « *critique* » et « *peine* », dans les exemples précédents), et la *valeur*, qui est choisie en fonction de la base (les mots « *virulente* » et « *purger* », dans les exemples précédents) et dont le sens est normalement idiomatique.

Dans ce travail de recherche, nous utilisons les formalismes du Web sémantique, dont les langages RDF et OWL, pour créer une ontologie métalinguistique appelée *lexfom* visant à encoder les fonctions lexicales. Cette ontologie est utilisée pour représenter les relations paradigmatiques et syntagmatiques présentes dans le *Réseau lexical du français* (RLF), qui est un réseau basé sur les fonctions lexicales dans un format d'ontologie informatique.

Nous avons extrait du RLF environ 600 fonctions lexicales standards, dont 100 fonctions simples et 500 fonctions complexes, 46 000 relations paradigmatiques et 8 000 collocations de la langue française, puis nous les avons encodées en utilisant notre modèle *lexfom*.

Mots clés : collocation, représentation des connaissances, ontologie, fonction lexicale, relation lexicale, Web sémantique, théorie Sens-Texte.

ABSTRACT

Collocations pose problems for several applications related to natural language processing, such as machine translation, information retrieval and natural language generationn. In particular, the representation of collocations remains an open and under-explored problem.

The lexical functions formalism has been developed to represent the many types of possible relations between words, such as the synonymy (*car - automobile*), antonymy (*big - small*) and hyponymy (*cat - mammal*), which are paradigmatic relations, and relations such as intensification (*harsh criticism*) and realization (*serve a sentence*), which are syntagmatic relations. The relation between the constituents of a collocation is of the syntagmatic type, which is modeled by the lexical functions of the same type.

In a lexical function relation, we can identify three parts : the *function* (such as synonymy, intensification or realization), the *base* or *keyword*, which is chosen freely and normally retains its prototypical meaning (words "*criticism*" and "*sentence*" in previous examples), and the *value*, which is chosen contingent to the base (words "*harsh*" and "*serve*" in preceding examples) and which normally has an idiomatic meaning.

In this thesis, we use the semantic web formalisms, the RDF and OWL languages, to create a meta-linguistic ontology called *lexfom* in order to encode the lexical functions. This ontology is used to represent the paradigmatic and syntagmatic relations present in the *French Lexical Network* (FLN), which is a network based on the lexical functions, in an ontology format.

We have extracted approximately 600 standard lexical functions from the RLF, including 100 simple functions and 500 complex functions, 46,000 paradigmatic relations and 8,000 collocations of the French language and we have encoded them using our *lexfom* model.

Keywords : collocation, knowledge representation, ontology, lexical function, lexical relation, semantic web, Meaning-Text Theory

INTRODUCTION

La représentation est une relation expliquée par l'association entre le *représenté* (objet, idée, événement, etc.) et le *représentant* (ce qu'on utilise à la place du représenté dans le cerveau ou dans une machine) (Pylyshyn, 1986).

C'est un sujet discuté depuis l'époque des philosophes grecs anciens. Par exemple, Aristote a défini la *substance* (*ousia*, en grec), une de ses dix catégories, comme l'essence même de chaque entité, qui est indépendante de ses propriétés ou des choses qu'on peut affirmer à son sujet. Brentano (1978) soutient, en regard des notions de substance et de représentation :

> We have seen that, according to Aristotle, the concept of substance is given directly in our perceptions, and that there cannot be any representation of an accident without that concept. Even when we apprehend ourselves as feeling or thinking, we apprehend ourselves as a feeling or thinking substance. (Brentano, 1978, p. 43)
>
> The general concept of the substance is always included in the representation of each accident, and this includes the accidents which are disclosed through inner perception. (*Ibid.*, p. 45)

Avant Aristote, Platon (Robin *et al.*, 1977) formule sa théorie selon laquelle les formes éternelles, qui se trouvent dans le monde des idées, sont les substances réelles, tandis que les objets du monde physique sont des ombres. Malgré qu'Aristote et Platon aient accordé plus d'importance à la notion de substance, ils parlaient déjà de la représentation de ces substances par notre perception.

La notion de représentation est bien présente dans notre vie quotidienne. On la

trouve dans plusieurs situations : les lettres de notre alphabet sont des représentations de sons. D'origine phénicienne, elles ont d'abord symbolisé des objets et des animaux, avant de prendre la forme des alphabets grec et latin que nous connaissons. Ainsi, la lettre *alf*, qui réfère à *boeuf*, est devenue la lettre *alfa* dans l'alphabet grec et la lettre *a* dans l'alphabet latin. Les panneaux de signalisation routière en sont un autre exemple : ils représentent des permissions, obligations, interdictions et directions. Chaque culture, chaque personne a ses propres gestes et ses icônes, qui peuvent représenter un même concept de différentes manières, ou même représenter différents concepts.

Cette relation entre le représentant et le représenté n'est pas univoque : un même objet peut être représenté par plusieurs représentants et un seul représentant peut représenter plusieurs objets. Dans les langues, cette non-univocité est à l'origine de la synonymie (lorsque deux ou plusieurs mots ont le même ou presque le même sens) et de la polysémie (lorsqu'un mot a deux ou plusieurs sens).

On peut représenter des choses concrètes (pierre, maison, arbre), des choses abstraites (émotions, désirs, croyances), des choses collectives (les populations d'un pays, l'ensemble des couleurs), des choses individuelles (une personne spécifique, une couleur), des équations mathématiques, des concepts physiques ($F = m \cdot a$, $E = mc^2$), des choses imaginaires (*le Père-Noël*, *Superman*), etc.

Cependant, comment l'être humain peut-il faire l'association entre le représentant et le représenté ? En sciences cognitives, différentes théories ont été proposées. Par exemple, les mentalistes, qui cherchent à comprendre le fonctionnement de l'esprit humain en s'appuyant sur l'introspection, affirment qu'un objet est représenté par son image, parce que sa représentation consciente mentale ressemble à l'objet (ressemblance) (Pylyshyn, 1986).

De son côté, le comportementalisme défend qu'une manifestation dans le cerveau

représente quelque chose si cette manifestation est semblable à l'événement qui se déroule quand la chose est perçue. La relation de représentation est expliquée par l'association, qui doit être activée pour la contiguïté et évoquée par l'activation d'autres termes associés (Pylyshyn, 1986).

D'autres chercheurs, comme Fodor (1975), ont développé la théorie appelée computationnalisme ou théorie computationnelle de l'esprit (*Computational Theory of Mind*) (Rescorla, 2015), selon laquelle le cerveau fonctionne comme un ordinateur.

Une des bases théoriques du computationnalisme est la thèse de Church-Turing (Kleene, 1952; Reus, 2016). Selon sa version mathématique, une machine de Turing (Turing, 1936), qui est l'équivalent mathématique de l'ordinateur moderne, peut faire tout ce qui est calculable (tout ce qu'un mathématicien peut faire en utilisant le calcul). Selon sa version physique, il est possible de simuler computationnellement n'importe quel système physique qui peut être représenté par une fonction mathématique calculable (Harnad, 2012).

La computation est une manipulation des symboles selon des règles (Turing, 1936). Par exemple, l'expression *+ 1 1 = 10* est une règle qui définit comment « transformer » la séquence de symboles *+*, *1* et *1* en une nouvelle séquence des symboles *10*. Le sens des symboles *+*, *1* et *0* n'importe pas, seulement leur forme : la computation est purement syntaxique.

Par forme, on veut dire que le symbole *1*, en étant différent du symbole *0*, va déclencher une règle différente ou donner un résultat différent. La forme est arbitraire : au lieu de *+*, on pourrait avoir *!*, et au lieu de *1* et *0*, on pourrait avoir *%* et *?*. Le sens de *+*, de *1* et de *0* est donné par des êtres qui font de la cognition, comme l'être humain (Harnad, 2012).

Il y a deux modèles principaux pour expliquer la représentation dans le computa-

tionnalisme : le modèle symboliste et le modèle connectiviste.

Le modèle symboliste est basé sur l'idée que, de la même façon que la computation, la cognition humaine n'est que manipulation de symboles, selon leurs formes, en suivant des règles. Pour Fodor (1975), il y a un langage de la pensée (*language of thought*) qui est syntaxiquement structuré et les processus mentaux sont définis sur ce langage.

Le modèle connectiviste (Elman *et al.*, 1996) est basé sur des modèles mathématiques, en particulier sur ceux appelés réseaux de neurones (RN). Un RN est inspiré par les réseaux neuronaux biologiques. Dans ce modèle, au lieu d'être représenté par un symbole, un objet du monde externe est représenté par les caractéristiques (*features*) qui le distinguent des autres objets.

Dans un RN, chaque unité du modèle (un neurone) ou groupes d'unités (groupe de neurones) se spécialise à reconnaître une des caractéristiques de l'objet. La représentation de l'objet « émerge » des connexions entre plusieurs unités (neurones). Par exemple, un RN peut « apprendre » à reconnaître des images de tables si on lui donne une quantité suffisante d'exemples.

En informatique, quand on parle de la représentation, on utilise plutôt le terme *représentation des connaissances* (RC) (*knowledge representation*). Davis *et al.* (1993) proposent cinq rôles pour expliquer ce qu'est la RC :

— La RC est un substitut (*surrogate*) : la représentation interne se substitue à des entités du monde externe ; au lieu de manipuler les vrais objets du monde réel, on manipule leurs représentations.

— Une RC est un ensemble d'engagements ontologiques (*ontological commitments*) : il n'existe pas de représentations parfaites. La seule représentation parfaite d'un objet serait l'objet lui-même. Pour construire une RC, il faut

décider quelles sont les caractéristiques d'un objet qu'on va ignorer, quelles sont celles qu'on va représenter, puis avec quel degré de précision nous allons les représenter.

— Une RC est une théorie fragmentée du raisonnement intelligent : typiquement, une RC ne contient qu'une partie de l'idée ou croyance qui l'a motivée. En plus, l'idée ou croyance originale n'est qu'une partie des différents types de raisonnement humain.

— Une RC est un moyen de réaliser un calcul efficace (*e cient computation*), ce pourquoi la RC est utilisée pour des tâches computationnelles. On s'attend à ce que les calculs nécessaires pour réaliser ces tâches soient efficaces.

— Une RC est un moyen d'expression humaine : elle permet d'informer une machine et de nous informer les uns les autres, en échangeant des informations à propos du monde.

Selon Kiefer (1988), on distingue trois types de connaissances : la connaissance linguistique, qui se réfère aux unités lexicales (lexies, expressions figées, etc.), la connaissance conceptuelle, qui concerne les catégories, indépendamment de leur lexicalisation, et la connaissance encyclopédique, qui est l'ensemble des informations sur les entités (réelles ou imaginaires), les actions, les événements, etc.

Nous allons illustrer ces trois types de connaissances en prenant l'exemple de la *table*. La connaissance encyclopédique de *table* rassemble tous les types d'informations liés à des tables en général : les différents formats, tailles, couleurs, prix, les lieux où les acheter, la quantité de tables dans notre propre maison et leur localisation, etc.

Le concept *table* inclut tous les objets qu'on utilise pour faire les choses qu'il est communément possible de faire avec les objets qu'on catégorise comme étant des *tables* (Harnad, 2010). Par exemple, on peut y déposer des objets, les reconnaître

comme *table*, les nommer comme tel, etc., ce qui les distingue de toutes les autres choses qu'on ne catégorise pas comme étant des *tables*. Le sens de *table*, par conséquent, est donné par la façon d'établir les membres et les non-membres de la catégorie *table*.

Finalement, le mot *table* est une étiquette arbitraire (Saussure *et al.*, 1972) qu'on attache à la catégorie des choses qu'on reconnaît comme étant des *tables*. La connaissance linguistique sur le mot *table* est la connaissance sur son utilisation dans une phrase ou expression, ses divers types de relations avec d'autres mots, ses différents sens, etc.

Dans cette thèse, nous nous intéressons à la représentation symbolique des connaissances linguistiques, et plus particulièrement aux relations entre les unités lexicales, connues comme relations lexicales ou sémantiques. Plus spécifiquement, nous souhaitons représenter les relations appelées *collocations*.

La définition de *collocation* que nous adoptons est celle de Mel'čuk *et al.* (1995), établie comme suit : une collocation est un phrasème sémantique, dont le sens est la composition du sens de ses deux unités lexicales constituantes. La première unité lexicale (la *base* ou *mot-clé*) est choisie librement. La seconde unité lexicale (le *collocatif*) est choisie en fonction de la base.

Voici des exemples de collocations : *faire attention*, *courir un danger*, *prendre en location*, *peur bleue*, *la situation s'aggrave*, *par téléphone*, *l'ouragan se calme*, *avis favorable*. À la section 1.9 nous donnons plus de détails sur la définition de collocation utilisée dans cette thèse.

Le formalisme linguistique utilisé pour modéliser les collocations est celui des *fonctions lexicales* (Mel'čuk *et al.*, 1995) de la théorie Sens-Texte (Žolkovskij et Mel'čuk, 1965), qui constitue la manière la plus complète et systématique de repré-

senter les collocations. Les fonctions lexicales modélisent les différentes relations sémantiques entre les unités lexicales, et non seulement les relations de collocation. Par exemple, entre les mots *grand* et *petit* il y a une relation d'antonymie, modélisée par la fonction lexicale *Anti* : Anti (*grand*) = {*petit*}. Les fonctions lexicales sont abordées au chapitre 4. Les principaux concepts de la théorie Sens-Texte sont présentés au chapitre 3.

Les formalismes informatiques adoptés pour encoder les fonctions lexicales sont ceux du Web sémantique, présentés au chapitre 6. Avec ces formalismes (les langages *RDF*/*RDFS* et *OWL*), il est possible de construire des représentations des connaissances qui peuvent être connectées sur le Web.

1. Problématique

L'être humain s'exprime par phrasèmes, c'est-à-dire par des expressions figées, des mots composés ou des expressions polylexicales, plutôt que par des mots séparés (Mel'čuk, 1998). Les collocations constituent l'absolue majorité des phrasèmes (*Ibid.*). Jackendoff (1997) affirme que la quantité de mots composés est aussi grande que la quantité de mots simples dans le lexique d'un locuteur natif.

Les collocations soulèvent des problèmes majeurs dans différentes applications du domaine du TALN, comme la traduction automatique, la recherche d'information et la génération de texte. Ramisch (2012) classifie la recherche sur les expressions polylexicales (EPL), dont les collocations sont un sous-groupe, en cinq tâches : extraction, identification, désambiguïsation, représentation et application des expressions polylexicales. Depuis Ramisch (2012), la représentation est considérée comme le thème de recherche le moins développé parmi ces tâches.

Différentes façons de représenter les collocations ont été proposées et cette représentation demeure encore aujourd'hui un problème ouvert. Nous passons en revue

les différentes méthodes pour les représenter, d'abord en utilisant les fonctions lexicales (FL), puis sans leur utilisation (au chapitre 7). L'extraction de collocations est aussi une tâche difficile et malgré qu'il ne s'agit pas d'un des objectifs principaux de cette thèse, nous présentons quelques méthodes en utilisant les FL pour l'extraction des collocations à la section 7.5. Les FL sont présentées au chapitre 4.

Un autre problème que nous souhaitons traiter est l'absence de la représentation de relations syntagmatiques dans la majorité des réseaux lexicaux, comme WordNet (Fellbaum, 1998) et FrameNet (Fillmore, 1977). Les relations syntagmatiques (sous-section 1.8.2) sont des relations entre unités lexicales, le même type de relation existant entre les mots dans une phrase ou entre les unités lexicales dans une collocation.

Les relations syntagmatiques sont importantes à cause de leur diversité et couverture. Par exemple, en français, la relation d'intensification est trouvée entre les paires lexicales suivantes, parmi d'autres :

— *camion lourd*, *très grave*, *aboyer férocement*, *ennemi juré*, *crime du siècle*, *ami proche*, *célibataire endurci*, etc.

Cette même relation d'intensification, par exemple, se trouve dans pratiquement toutes les langues et entre un grand nombre d'unités lexicales. Cependant, cette relation est ignorée dans la majorité des réseaux lexicaux existants. Et la relation d'intensification n'est qu'une relation parmi des dizaines d'autres relations syntagmatiques absentes des réseaux lexicaux.

La majorité des réseaux lexicaux ne représentent que des relations paradigmatiques entre mots (synonymie, antonymie, hyperonymie, hyponymie, etc.) (Schwab *et al.*, 2007). Quelques solutions ont été proposées pour régler cette déficience,

comme celle de Bentivogli et Pianta (2004) et celle de Schwab *et al.* (2007). Cependant, ces relations ne sont pas représentées d'une manière si systématisée et englobante, en comparaison aux FL.

D'un point de vue cognitif, les réseaux lexicaux relationnels sont vus comme des représentations du lexique mental (Grossmann, 2011). Par exemple, la motivation derrière la création de WordNet est liée aux travaux psycholinguistiques de Quillian (1968) et de Fellbaum (1998). Toutefois, comme il manque des relations syntagmatiques dans la majorité des réseaux lexicaux, il manque également une quantité importante de relations sémantiques entre unités lexicales.

Au-delà des réseaux lexicaux, les relations lexicales jouent un rôle important dans plusieurs tâches liées au TALN, comme la génération (Wanner et Bateman, 1990; Lambrey et Lareau, 2015) et la désambiguïsation de texte (Navigli, 2009).

Par exemple, les méthodes les plus performantes pour la désambiguïsation de texte sont basées sur des relations lexicales trouvées dans des graphes lexicaux (Moro et Navigli, 2015). Cependant, ces méthodes ne sont basées que sur des relations paradigmatiques. Nous croyons que l'enrichissement et l'utilisation des réseaux lexicaux, qui représentent aussi des relations syntagmatiques, pourraient donner des résultats encore plus performants.

En relation à la sémantique distributionnelle, Sahlgren (2006) montre l'importance de la représentation des relations paradigmatiques et syntagmatiques dans des vecteurs de mots. Cependant, cette notion de relation syntagmatique est simplement celle de la cooccurrence statistique ; elle ne prend pas en considération le raffinement et la systématisation de telles relations, tels que modélisés par les FL.

Un dernier problème est le manque d'une représentation informatique des FL, c'est-à-dire, d'une représentation des propriétés et des caractéristiques des FL, de

leurs classifications syntaxique et sémantique. Les FL sont le seul outil linguistique à modéliser d'une manière systématique les relations syntagmatiques.

Nous croyons que le formalisme de représentation des FL et des collocations que nous présentons dans cette thèse sera utile pour l'enrichissement d'un réseau lexical et pour la résolution du problème du manque de relations syntagmatiques dans les réseaux lexicaux en général.

2. Objectifs de la recherche

L'objectif principal de cette thèse est la proposition d'une méthode originale pour la représentation des collocations. L'approche adoptée est de les représenter dans un réseau lexical en utilisant les fonctions lexicales et les formalismes du Web sémantique.

Nous croyons que représenter les collocations à l'aide de liens entre unités lexicales dans un graphe lexical est une manière plus intuitive d'y arriver, au lieu de, par exemple, créer des tableaux pour représenter les mots simples et les mots composés, tel que proposé par Villavicencio *et al.* (2004).

En plus, comme nous l'avons déjà mentionné, les RL sont vus comme des représentations du lexique mental. Il nous semble donc plus cohérent, d'un point de vue cognitif, de représenter des relations lexicales telles que les collocations dans un réseau lexical, au lieu de les représenter dans des tableaux ou des grammaires formelles. Notre modèle de représentation de collocations pourrait être utilisé dans d'autres réseaux lexicaux et sémantiques, car le format des graphes permet une connexion plus facile entre les ressources de ce type.

Le deuxième objectif est le développement d'une ontologie métalinguistique pour la représentation des fonctions lexicales de la théorie Sens-Texte. Cette ontologie pourrait être utilisée par la communauté scientifique qui travaille sur le dévelop-

pement et l'enrichissement des ontologies, spécialement les ontologies lexicales, pour la création de liens syntagmatiques entre unités lexicales et l'enrichissement des liens paradigmatiques, tel qu'expliqué à la section 1.8.

Cette ontologie métalinguistique sera compatible avec *lemon* (McCrae *et al.*, 2011), qui est présentement la recommandation de W3C pour la représentation des informations linguistiques sur le Web sémantique.

Le choix de construire une ontologie linguistique en utilisant les formalismes du Web sémantique (les langages *RDF*/*RDFS*/*OWL*) se justifie par le succès que ces formalismes ont obtenu ces dernières années. Par exemple, Hendler et van Harmelen (2008) affirment qu'ils sont les langages de représentation des connaissances les plus utilisés dans l'histoire.

De plus, des représentations basées sur le langage RDF forment des graphes de connaissances, ce qui se combine bien avec une représentation lexicale sous la forme d'un graphe, comme un réseau lexical. Finalement, il existe déjà plusieurs représentations des connaissances basées sur ces formalismes, y compris des réseaux lexicaux comme WordNet, auxquels notre modèle peut s'intégrer.

Le troisième objectif de cette thèse est la transformation du Réseau lexical du français (RLF) (Lux-Pogodalla et Polguère, 2011) de son format actuel vers un format compatible avec les formalismes du Web sémantique. Le RLF est, à notre connaissance, le seul réseau lexical basé sur les fonctions lexicales de la théorie Sens-Texte et le seul à avoir implémenté des relations syntagmatiques d'une manière systématique et ample. Cependant, le RLF est présentement dans un format de base de données relationnelle et notre objectif est de le représenter dans un format qui soit compatible avec des connaissances représentées par des ontologies informatiques.

D'autres représentations basées sur les fonctions lexicales existent, comme DiCE (Alonso Ramos, 2006), DiCoInfo (L'Homme, 2008) et DiCoEnviro (L'Homme et Lanneville, 2014). Cependant, ce sont des dictionnaires plutôt que des réseaux lexicaux.

Finalement, un dernier objectif est l'implémentation d'une toute première classification sémantique des fonctions lexicales en tant qu'ontologie métalinguistique. Les fonctions lexicales peuvent être regroupées de plusieurs façons. Par exemple, Jousse (2010) présente quatre classifications possibles, selon leurs propriétés syntaxiques, sémantiques, combinatoires et pragmatiques.

Dans cette thèse, nous présentons un module de notre modèle qui implémente la classification sémantique proposée par Jousse (2010). Ce module permettra de connecter des sens prédicatifs aux représentations des fonctions lexicales, en donnant la possibilité, par exemple, de faire l'extraction des collocations modélisées par des fonctions qui ont un sens prédicatif déterminé (par exemple, extraire des collocations modélisées par des fonctions lexicales dont le sens prédicatif est l'*intensification* ou extraire seulement des collocations formées par des *verbes supports*, etc.).

3. Structure du document

Cette thèse est structurée comme suit : Au chapitre 1, nous présentons les fondements théoriques liés aux aspects cognitifs de cette thèse : les notions de sens, de concept, de catégorie, de mot, de langue, de lexique, de relation lexicale et de collocation.

Au chapitre 2, nous présentons les principaux courants liés à la représentation linguistique et lexicale, en soulignant les préceptes de la linguistique cognitive.

Au chapitre 3, nous présentons les aspects de la théorie Sens-Texte sur lesquels

les fonctions lexicales sont basées et qui sont les plus pertinents pour notre thèse.

Au chapitre 4, nous présentons les fonctions lexicales, les différentes manières de les classifier ainsi que leurs propriétés, avantages et possibles applications informatiques.

Au chapitre 5, nous présentons la définition de réseau lexical, sa motivation et quelques exemples de réseaux lexicaux.

Au chapitre 6, nous présentons le Web sémantique, ses formalismes et la définition d'une ontologie au sens informatique, y compris des ontologies métalinguistiques comme *lemon*.

Au chapitre 7, nous passons en revue les différents modèles de représentation d'expressions polylexicales et de collocations, d'abord en utilisant les fonctions lexicales, puis sans leur utilisation. Aussi, nous présentons des travaux sur l'extraction des collocations à partir de corpus textuels en utilisant les fonctions lexicales.

Au chapitre 8, nous présentons *lexfom*, l'ontologie métalinguistique que nous avons développée pour la représentation de fonctions lexicales et de collocations. Nous montrons aussi un exemple de comment *lexfom* est utilisée pour la représentation d'une collocation et comment nous avons transposé le RLF dans un format compatible avec les formalismes du Web sémantique.

Au chapitre 9, nous montrons une application pratique de notre implémentation du Réseau Lexical du Français sous la forme d'une ontologie lexicale.

En conclusion, nous présentons les contributions de cette thèse et les perspectives qu'elle ouvre dans le domaine de la représentation des connaissances linguistiques.

CHAPITRE I

FONDEMENTS THÉORIQUES

1.1 Introduction

Après avoir présenté une introduction à notre travail de recherche, la problématique et les objectifs de cette thèse, nous présentons dans ce chapitre les fondements théoriques de notre démarche et les définitions que nous avons utilisées.

1.2 Concept

La notion de *concept* est définie de plusieurs façons dans la littérature. Par exemple, Margolis et Lawrence (2005) citent trois compréhensions différentes du terme :

i) Le concept comme représentation mentale (*mental representation*) : les concepts sont conçus comme des entités psychologiques et la pensée se passe dans un système interne de représentation. Par ailleurs, à chaque attitude propositionnelle (croyance, désir) correspond une représentation mentale interne.

ii) Le concept comme capacité (*ability*) : selon cette conception, les concepts ne sont pas des images mentales ni des entités en forme de mots dans le langage de la pensée, mais plutôt des habiletés propres des agents cognitifs.

iii) Le concept au sens frégéen (*Fregean sense*) (Frege, 1948) : les concepts sont identifiés comme des objets abstraits, au lieu d'états mentaux ou d'objets

> mentaux. Ils servent d'intermédiaires entre la langue et la pensée, d'un côté, et les référents, d'un autre côté.

Nous développons maintenant la notion de concept basée sur la théorie des graphes conceptuels de Sowa (1984).

Ce dernier définit la perception comme un processus de construction d'un modèle pour la représentation des entrées sensorielles. Le modèle contient deux parties : une sensorielle, formée par une interconnexion de percepts (à chaque percept correspond un aspect de l'entrée), et une partie abstraite, le graphe conceptuel, qui décrit comment les percepts s'intègrent pour former le concept.

Les concepts concrets sont associés directement à des percepts. Les concepts abstraits ne le sont pas. Ils sont plutôt associés à un réseau de relations entre concepts qui, à la fin, les connectent à un concept concret et, par conséquent, à des percepts.

L'ensemble des relations entre concepts, percepts et procédures est appelé réseau sémantique (*Ibid.*).

Un graphe conceptuel (GC) est une représentation par graphe de la logique de premier ordre basée sur les réseaux sémantiques de l'intelligence artificielle et les graphes existentiels de Charles Peirce (*Ibid.*). Ils constituent un langage pour la représentation des connaissances, où des nœuds conceptuels représentent des entités, des attributs, des états et des événements. Des nœuds relationnels font l'interconnexion entre les concepts (*Ibid.*).

Les GC ont évolué à partir des graphes existentiels (GE) de Peirce (Searle *et al.*, 1997). Les GE ont été conçus pour la représentation de la logique et pour traiter des propositions logiques, en utilisant les primitives les plus simples possible. Par contre, l'objectif des GC est de faire le lien le plus direct possible entre les langues naturelles et la logique (*Ibid.*).

Depuis sa proposition par Sowa (1984), la théorie des graphes conceptuels a évolué et plusieurs types de graphes ont été créés avec l'objectif d'améliorer la représentation des contextes ou la représentation des connaissances en général.

Il est intéressant de noter le lien entre les graphes conceptuels et les technologies actuelles pour la représentation des connaissances sur le Web, comme les langages *OWL* et *RDF*, via Common Logic (Sowa, 2008, 2010).

Nous détaillons dans les sous-sections suivantes quelques notions importantes des GC : le contexte, les relations conceptuelles, le percept et le réseau sémantique.

1.2.1 Contexte

Le *contexte* est l'idée centrale dans la théorie de Peirce et, par conséquent, dans la théorie des GC. D'après Sowa (1997), les deux sens principaux du mot *contexte* existent dans les dictionnaires :

— au sens basique : il s'agit alors d'une section de texte ou d'un discours autour d'un mot ou d'une phrase ;

— au sens dérivé : il s'agit alors d'une situation non linguistique, soit l'environnement, le domaine ou le fond dont une entité ou un sujet fait partie.

Le travail de Peirce a créé la base unifiante des idées sur la représentation du contexte qui se sont disséminées parmi les recherches modernes sur la logique, la linguistique et l'intelligence artificielle (Sowa, 1997), y compris dans le cas des graphes conceptuels.

1.2.2 Relation conceptuelle

Une *relation conceptuelle* est une connexion entre concepts. Nous pouvons la classifier en deux types (Kavalec et Svátek, 2005) : la relation taxonomique et la relation non taxonomique.

Le premier type est formé par les relations qui divisent les entités en groupes, pour former une taxonomie (classification d'un type déterminé d'entités, comme la taxonomie des animaux vertébrés : mammifères, oiseaux, reptiles, poissons, amphibiens). Ces relations sont de type *est-un* (*is-a*) (*un chien est-un mammifère*) et de type *partie-de* (*part-of*) (*le moteur fait partie de la voiture*). La relation taxonomique est connectée à celle de *subsomption* : par exemple, la notion de mammifère subsume celle de chien.

Le second type est formé par les relations non taxonomiques (celles qui ne forment pas une classification) entre les concepts. Par exemple, il y a plusieurs relations entre une compagnie et un produit : une compagnie fabrique un produit, elle vend un produit, elle transforme un produit, etc.

Les relations taxonomiques sont fondamentales pour la construction et le développement des ontologies au sens informatique, y compris les ontologies lexicales.

1.2.3 Percept

Un *percept* est une petite information captée par nos capteurs externes : des goûts, des images, des sons, etc. Ils sont groupés pour former des concepts (Sowa, 1984). Certains graphes conceptuels décrivent comment les percepts sont combinés.

Des relations conceptuelles spécifient le rôle de chaque percept. Par exemple, le percept d'une couleur peut être combiné celui d'une forme pour composer un graphe qui représente une forme colorée.

Il est possible que les percepts les plus simples et les plus basiques soient les points d'ancrage entre les mots (la séquence de caractères, les signifiants) et leurs sens (les signifiés). Autrement dit, les sens des mots sont ancrés à la nature à partir de nos capteurs externes (Harnad, 1990). Nous pouvons aussi établir une relation entre les percepts les plus simples et les primitifs sémantiques introduits par Wierzbicka

(Wierzbicka, 1996; Valente, 2002).

1.2.4 Réseau sémantique

Dans la théorie des GC, un *réseau sémantique* est l'ensemble de toutes les relations des concepts avec d'autres concepts, percepts et mécanismes moteurs (Sowa, 1984).

Quelques concepts dans le réseau sémantique sont ancrés directement à des percepts, qui sont de nature physique ou concrète, tandis que d'autres tirent leur sens des entités (autres concepts, percepts, relations entre concepts, etc.) dans le réseau sémantique.

Par exemple, le concept de *propriété* n'est pas ancré directement à un percept physique ; il tire son sens d'une interconnexion de concepts, relations, etc.

Un GC fait partie d'un réseau sémantique et il tire son sens de sa connexion avec lui. Le réseau sémantique relie un GC à un contexte, à des émotions, percepts, règles de grammaire, etc.

1.3 Catégorie

Une *catégorie* est une partition de la réalité. Cette partition peut être arbitraire, motivée par des nécessités de survie (nécessités alimentaires, d'habitation, etc.), motivée par la nécessité de mieux comprendre un domaine ou un sujet, etc.

Le premier système catégoriel connu a été proposé par Aristote. Il a affirmé que tout ce qu'il y a dans la nature peut être divisé en dix catégories (Ackrill, 1975) : la *substance* (*essence*), la *quantité*, la *qualité*, la *relation*, le *lieu*, le *temps*, la *position*, la *possession*, l'*action* et la *passion*.

En suivant les idées d'Aristote, Kant a proposé une table de douze catégories,

organisées en quatre groupes (Eco, 1999, p. 77) :

— quantité : unité, pluralité, totalité ;

— qualité : réalité, négation, limitation ;

— relation : substance et inhérence, causalité (cause / effet), communauté (action réciproque agent / patient) ;

— modalité : possibilité / impossibilité, existence / non-existence, nécessité / contingence.

Lakoff (1993) utilise la métaphore d'un récipient pour parler de la catégorie : on peut mettre quelque chose dans une catégorie ou retirer quelque chose d'une catégorie, etc. Les propriétés d'un récipient s'appliquent à des catégories : par exemple, si A est dans le récipient B et B est dans le récipient C, alors A est dans le récipient C.

Selon Lakoff (1993), les propriétés logiques des catégories peuvent être comprises comme un héritage des propriétés logiques des récipients, auquel s'ajoute le lien métaphorique entre les récipients et les catégories. Cette approche métaphorique pour expliquer la notion de catégorie fait partie de la théorie appelée *schéma d'image* (*image schema*) (Johnson, 1990; Lakoff, 1991), qui est présentée dans la première partie de la sous-section 2.3.2.

La catégorisation et la perception catégorielle sont liées au langage. Par exemple, Hirst (2004) considère chaque sens d'un mot comme une catégorie, prenant en considération que le sens d'un mot garde un caractère flou et subjectif, comme plusieurs catégories du monde physique. Par contre, tel que mentionné par Hirst (2004), les critiques de cette vision soulignent que le sens d'un mot est dérivé, créé et modulé selon l'usage et le contexte et qu'il ne peut pas, du fait, faire partie du lexique d'une langue.

L'un des mécanismes que nous pouvons utiliser pour acquérir les catégories est la *perception catégorielle*. Prenons les catégories chromatiques : le spectre des couleurs est continu. Cependant, pour identifier des catégories, il faut identifier des pôles (*rouge*, *bleu*, *jaune*, etc.), autour desquels il y a une concentration de notre perception (Cibelli *et al.*, 2016).

Les fréquences de la lumière qui sont plus proches du « rouge », nous les voyons toutes comme étant la couleur rouge. La perception catégorielle fait qu'il y a une compression dans une catégorie et une séparation entre les catégories (Cibelli *et al.*, 2016). Par exemple, nous voyons deux tonalités de rouge, *rouge1* et *rouge2*, comme étant plus proches l'une de l'autre que la tonalité *rouge2* de la tonalité *jaune1*, même si la différence, en termes de fréquence, est la même (Harnad, 2005; Cibelli *et al.*, 2016).

Cette perception catégorielle pour les couleurs n'est pas innée et on pourrait dire quelle est une sorte d'effet de Sapir-Whorf (Lucy, 1998), l'hypothèse soutenant que notre perception du monde physique est influencée par le langage. La catégorisation des animaux, par exemple, est différente, parce qu'il n'y a pas un continuum entre les animaux. Par exemple, il y a des lacunes entre un lion et un zèbre (nous n'avons pas besoin de faire la compression, la nature l'a déjà faite pour nous).

Encore dans l'esprit de l'effet de Sapir-Whorf, citons les mots d'Eco (1999) :

> Parler de ce qui est, cela veut dire rendre communicable ce que nous en connaissons ; mais l'acte même de connaître, et de communiquer, implique le recours au générique. Le générique est déjà un effet de la sémiose et dépend d'une segmentation du contenu dont le système kantien des catégories, attaché à une vénérable tradition philosophique, est un produit culturel déjà organisé, enraciné culturellement et ancré linguistiquement (Eco, 1999, p. 70).

Finalement, lorsque nous apprenons une catégorie, nous pouvons la nommer, lui attacher une étiquette (mot) (Saussure *et al.*, 1972). Et on peut regarder les sens des mots comme des catégories (Hirst, 2004). Par exemple, la catégorie *éléphant* sépare les animaux entre ceux qui sont des éléphants et ceux qui ne le sont pas (Harnad, 2005). Même les noms propres peuvent être considérés comme des catégories. *John Smith* est la catégorie de la personne *John Smith* en différentes positions, avec différents vêtements, en différents moments. Nous devons oublier toutes ces variations pour faire l'abstraction de ce qui est *John Smith* (Borges, 1981).

1.4 Sens

Il y a plusieurs théories et courants qui expliquent ce qu'est le *sens* d'un mot. Lyons (1995) cite les six théories suivantes :

i) la théorie référentielle ou dénotationnelle : le sens d'une expression ou d'un mot est celui auquel l'expression ou le mot fait référence. Par exemple, le sens du mot *chien* dans une phrase est un chien spécifique, dont le locuteur parle, ou la catégorie qui représente tous les chiens. Cela dépend du contexte ;

ii) la théorie mentaliste : le sens d'un mot est associé à une idée ou à un concept dans l'esprit de celui qui l'a exprimé ;

iii) la théorie comportementaliste : le sens d'un mot est le stimulus qui l'évoque ou la réponse évoquée par lui ;

iv) la théorie sens-est-utilisation : le sens d'un mot est déterminé par, ou même identique à, son usage ;

v) la théorie vérificationniste : le sens d'une expression est déterminé par la vérifiabilité des propositions qui la contiennent ;

vi) la théorie vérité-conditionnelle : le sens d'une expression est déterminé par

sa contribution aux conditions de vérité des propositions qui la contiennent.

Dans cette thèse, nous considérons comme sens une combinaison des théories *i*, *ii* et *iv*.

Le sens d'un mot est référentiel : il connecte le mot à une entité, événement, action, etc. En même temps, ce lien a un point de départ, qui est l'esprit du locuteur. Ce dernier point est important, car on assume qu'il y a une relation entre sens et communication, tel qu'exprimé par Lyons (1995) et défendu par Wittgenstein (1953). C'est aussi un point important de la théorie Sens-Texte, sur lequel nous reviendrons au chapitre 3.

Finalement, nous soulignons la vision défendue par Harnad (1990), qui n'est pas mentionnée par Lyons (1995), mais qui est liée aux points *i* et *iv* ci-dessus : le sens d'un mot est celui qui sépare les membres et les non membres d'une catégorie.

Illustrons cela avec un exemple : dans une tribu, les collecteurs de champignons apprennent à différentier les champignons comestibles des non comestibles. Il n'y a pas encore de mots pour définir ces deux catégories. Comme les champignons sont importants pour cette tribu, pour éviter de dire tout le temps « les champignons comestibles » ou « les champignons non comestibles », ils ont créé le mot *sofo* pour désigner les champignons comestibles et le mot *sifi* pour l'autre groupe. Si on demande : « quel est le sens du mot *sofo* ? », on peut l'expliquer avec des mots ou on peut simplement montrer tous les champignons, les séparer en deux catégories, *sofo* et *sifi* et montrer avec des gestes que les *sofo* sont comestibles et les *sifi* ne le sont pas.

Finalement, il faut souligner que le sens d'un mot dépend du contexte où il est utilisé et de ses relations avec d'autres mots (relations lexicales ou sémantiques) qui font partie du lexique d'une langue.

À la section 2.3.2, nous présentons différentes théories de la linguistique cognitive pour expliquer le sens.

1.5 Mot, vocable, lexie, lexème et phrasème

Tel que déjà mentionné, Saussure *et al.* (1972) ont défini le *mot* comme une étiquette arbitraire. Et tel que cité à la section précédente, cet étiquetage est fait pour nommer des catégories.

Selon Lyons (1995), un mot est une unité composite, constituée d'une forme et d'un sens.

Dans cette thèse, nous séparons la forme et le sens d'un mot en utilisant les définitions de *vocable* et *lexie* de la théorie Sens-Texte.

Un vocable regroupe différents mots qui se différencient les uns des autres seulement par leurs formes fléchies. Par exemple, *manger*, *mange*, *mangeront*, *mangeais*, etc., ne sont que des variations d'un seul vocable, représentées par leur forme canonique MANGER (Mel'čuk *et al.*, 1995).

Selon Mel'čuk *et al.* (1995), une « lexie est un mot pris dans une seule acception bien spécifique (lexème) » ou « une locution prise dans une seule acception bien spécifique (phrasème) » et qui est « muni de tous les renseignements qui spécifient totalement son comportement dans un texte ».

« Pris dans une seule acception bien spécifique » signifie qu'un vocable peut représenter plusieurs lexèmes et chaque lexème représente un seul sens du vocable (ou expression), selon le contexte. Par exemple, pour le vocable PONT nous avons, parmi d'autres, les lexèmes suivants :

— Lexème 1 : construction reliant les deux rives d'une étendue d'eau ;

— Lexème 2 : jours chômés entre deux jours fériés ;

— Lexème 3 : pont (aérien), liaison régulière par avion entre deux points, etc.

Chacun de ces lexèmes du vocable PONT est un sens de ce vocable. Dans cette thèse, nous utilisons l'expression *unité lexicale* pour parler indistinctement d'un lexème ou d'un phrasème. Les relations lexicales (en réalité, ce sont plutôt des relations sémantiques) que nous encodons pour représenter des collocations sont des relations entre des unités lexicales (lexèmes et phrasèmes) plutôt que des relations entre vocables.

Nous prenons en compte la représentation des unités lexicales dites pleines (*full forms*), (Lyons, 1995), qui sont les unités qui appartiennent aux classes des noms, adjectifs, verbes et adverbes. Les autres formes, dites *mots fonctionnels* (prépositions, conjonctions, articles, etc.), apparaissent dans la représentation du régime d'une collocation, tel qu'expliqué à la section 1.9. Cependant, certaines prépositions et conjonctions font partie d'unités lexicales pleines. Par exemple : *sur le lit*, *dans la voiture*, etc.

Dans cette thèse, nous utilisons les conventions typographiques suivantes, basées sur les conventions adoptées par la TST (Mel'čuk *et al.*, 1995, p. 13) :

1. Les exemples linguistiques sont imprimés en caractères italiques.
2. Les noms de lexies (vocables) sont imprimés en capitales.
3. Les premières mentions de termes importants sont imprimées en caractères italiques.

De plus, les mots et les expressions dans des langues autres que le français sont imprimés en caractères italiques.

1.6 Langue

Une *langue* est un système de signes doublement articulé, utilisé comme moyen de communication entre les membres d'une communauté (Coseriu, 1986). « Doublement articulé » signifie que la langue est formée de deux niveaux : le niveau des morphèmes, qui peuvent être des mots ou des parties de mots, comme les racines, les préfixes, les suffixes, etc. ; et le niveau des phonèmes, qui sont les sons de la langue.

Alors que les morphèmes portent du sens, les phonèmes n'en portent pas. Pour chaque langue, ce sont les règles de morphologie, de syntaxe, de sémantique et combinatoire, qui gouvernent la manière dont les phonèmes sont organisés en morphèmes. C'est cette double articulation qui donne leur versatilité aux langues humaines, car elle permet qu'une quantité finie de sons puisse être combinée pour former une quantité pratiquement infinie de sens.

La linguistique a connu ses débuts comme science moderne avec la publication des notes de classes de Ferdinand de Saussure par ses étudiants, notamment de ses fameuses dichotomies *langue*/*parole*, *signifié*/*signifiant*, *syntagme*/*paradigme*.

Pour Saussure *et al.* (1972), le langage est un système de signes qui exprime des idées. C'est un système formel divisé en deux parties : la *langue*, qui est la partie abstraite du langage internalisée par une communauté, et la *parole*, qui est l'acte individuel de parler et la réalisation pratique de la langue.

Saussure était intéressé par la langue qui, selon lui, est homogène et générale, en opposition la parole, qui est hétérogène et individuelle. L'unité basique de la langue est le *signe*. Le signe est formé par le *signifiant*, son image vocale, et par le *signifié*, qui est l'idée abstraite, mentale, évoquée par le signifiant. Les deux parties du signe sont inséparables, comme les deux côtés d'une feuille de papier.

Les caractéristiques du signe linguistique sont les suivantes :

— Le signe est arbitraire : il n'y a rien dans la séquence de caractères « maison » ou dans les sons produits quand on prononce « maison » qui évoque l'idée abstraite, le signifié, qu'on associe normalement à l'image mentale d'une maison. Cette relation entre signifiant et signifié est arbitraire : elle est définie par convention, pour des raisons historiques ou culturelles.

— Le signe a une *valeur* : la valeur d'un signe est déterminée par tous les autres signes de la même langue. La valeur vient de la comparaison, de l'opposition ; il y a donc des relations entre les signes. Pour Saussure, ces relations sont de deux types : *syntagmatiques* et *paradigmatiques*.

Les relations syntagmatiques et paradigmatiques sont centrales à la présente thèse et seront expliquées plus en détail à la section 1.8.

1.7 Lexique

On peut définir le lexique comme « l'ensemble des mots qu'une langue met à la disposition des locuteurs » (Picoche, 1977). Mel'čuk *et al.* (1995) appellent lexique d'une langue l'ensemble de toutes les lexies de cette langue.

Différents courants linguistiques proposent des modèles pour la description du lexique. Nous présentons au chapitre 2 ces différents courants et au chapitre 3 la théorie Sens-Texte.

Historiquement, le lexique des langues de tradition écrite est représenté dans un dictionnaire. Selon Lyons (1995), le lexique est l'équivalent théorique d'un dictionnaire. Cet auteur ajoute :

> Looked at from a psychological point of view, the lexicon is the set (or network) of all the lexemes in a language, stored in the brains of

> competent speakers, with all the linguistic information of each lexeme that is required for the production and interpretation of the sentences of the language (Lyons, 1995, p. 73).

Dans cette thèse, nous optons pour représenter le lexique comme un réseau lexical, au lieu d'utiliser une représentation du dictionnaire. Différents réseaux lexicaux sont présentés à la section 5, y compris le RLF, que nous utilisons pour démontrer notre méthode de représentation des collocations.

Dans un réseau lexical, on s'intéresse davantage aux relations entre unités lexicales qu'à leur définition. Dans la prochaine section, nous décrivons les deux types de relations lexicales concernés par notre thèse : les relations paradigmatiques et les relations syntagmatiques.

1.8 Relation lexicale

Une relation lexicale, aussi connue sous le nom de relation lexico-sémantique, est une relation récurrente entre deux unités lexicales, par exemple la relation de synonymie entre *voiture* et *automobile* ou celle de méronymie (partie de) entre *doigts* et *main*.

Les relations lexicales ont plusieurs propriétés : leur *nature* (*conceptuelle* vs *contextuelle*), leur *caractère prototypique* vs leur *caractère idiomatique*, etc. (Jousse, 2010). Dans cette thèse, nous ne nous intéressons qu'à une des propriétés, la *dimension paradigmatique* vs la *dimension syntagmatique*, qui est une des dichotomies présentées par Saussure *et al.* (1972).

1.8.1 Relation paradigmatique

Une relation paradigmatique entre lexies (ou entre parties d'une lexie) est connue comme relation verticale, associative ou *in absentia* (Saussure *et al.*, 1972). Par

exemple, dans la phrase : « le garçon a pris la chemise », il y a des relations paradigmatiques entre chaque lexie et d'autres lexies qui pourraient les remplacer. Dans cette phrase, la lexie *chemise* est en relation paradigmatique avec *pantalon*, *ballon*, *argent*, car elle peut être remplacée par ces autres lexies (« le garçon a pris le pantalon », « le garçon a pris l'argent », etc.).

De plus, on peut dire qu'il y a des relations paradigmatiques entre *garçon* et *homme*, *femme*, *enfant*, etc., qui représentent des lexies qui pourraient le remplacer dans cette phrase. En revanche, on ne pourrait pas dire qu'il y a une telle relation entre *garçon* et *chaise*, par rapport à cette phrase, car une *chaise* ne peut pas « prendre une chemise ».

Cependant, si la phrase est « le garçon est devant la table », alors il y a une relation paradigmatique entre *garçon* et *chaise*, par rapport à cette phrase, car une chaise peut être aussi devant une table.

Finalement, par rapport à la première phrase, on peut dire qu'il y a une relation paradigmatique entre *pris* et *acheté*, *vendu*, etc.

Les exemples précédents présentent des relations paradigmatiques dites « de parole ». Dans cette thèse, nous ne nous intéressons que par les relations paradigmatiques dites « de langue », parmi lesquelles les principales sont : la *synonymie*, l'*antonymie*, l'*hypero-nymie* et la *méronymie*. Il y a un lien entre ce type de relation et les relations taxonomiques (Section 1.2.2), qu'on utilise pour catégoriser le monde. Par exemple, il y a une relation paradigmatique entre « *garçon* » et « *chien* » par rapport à la phrase « *le garçon a mangé sa nourriture* », parce que « *garçon* » et « *chien* » appartiennent à une même catégorie, celle des êtres qui sont capables de manger.

1.8.2 Relation syntagmatique

Une relation syntagmatique est une relation horizontale ou *in praesentia* (Saussure *et al.*, 1972) entre lexies ou dans une lexie. Dans la phrase « Les étudiantes parlaient à leurs parents. », il y a des relations syntagmatiques entre *les* et *étudiantes*, *étudiantes* et *parlaient*, *parlaient* et *leurs parents*, *étudiantes* et *parents*, etc. Les relations syntagmatiques sont des relations non taxonomiques (Section 1.2.2).

Il y a aussi des relations syntagmatiques entre *le* et *s* pour former *les*, entre *étudiant* (racine), *e* (marque du féminin) et *s* (marque du pluriel), etc. Un syntagme est toujours formé de deux ou plusieurs unités (mots, morphèmes, etc.) consécutives et, dans un syntagme, chaque partie obtient son sens à partir de l'opposition à la partie précédente ou suivante (Saussure *et al.*, 1972).

Si, par exemple, on prend la collocation *faire une erreur*, il y a une relation syntagmatique entre *faire* et *erreur*. Dans la section suivante, nous présentons la définition de collocation utilisée dans cette thèse.

1.9 Collocation

Une *collocation* est un type spécifique d'expression polylexicale (EPL) (ou mots composés, *i. e.* des expressions formées par deux ou plusieurs mots).

La définition exacte de *mot composé* ou d'expression polylexicale varie selon l'auteur. Selon Moon (1998) :

> [...] there is no unified phenomenon to describe but rather a complex of features that interact in various, often untidy, ways and represent a broad continuum between non-compositional (or idiomatic) and compositional groups of words (Moon, 1998).

De plus, ce phénomène reçoit différents noms dans la littérature (Proost, 2007) : unités phraséologiques, expressions figées, combinaisons de mots, phrasèmes, etc.

Dans cette thèse, nous considérons une EPL d'une manière similaire à ce que Mel'čuk (1998) définit comme un phrasème : une expression qui n'est pas libre, c'est-à-dire que le signifié et/ou le signifiant de l'expression n'est pas construit sans restrictions ou d'une manière régulière.

Selon Saussure *et al.* (1972), un signe linguistique est formé par deux parties inséparables : le *signifié* et le *signifiant*. Le signifié est une représentation mentale de la dénotation du signe, définie par les caractéristiques qui distinguent ce signe de tous les autres signes de la langue. Le signifiant est l'image acoustique mentale de l'aspect matériel du signe.

Une phrase est construite sans restriction lorsque les règles utilisées dans sa construction ne sont pas obligatoires. Par exemple, au lieu de dire « se rendre au travail en voiture », on pourrait dire « se rendre au boulot en auto » ou « aller à la job en char ».

Par contre, une expression comme *fièvre de cheval*, qui a le signifié *forte fièvre*, est figée. Il n'existe pas d'expressions équivalentes comme **fièvre d'ours*, **fièvre de vache*, **hyperthermie de cheval* ou **toux de cheval*, même si elles sont grammaticalement correctes. Par conséquent, l'expression *fièvre de cheval* est un phrasème, parce sont signifié n'est pas construite sans restriction.

Selon Mel'čuk (2015), il y a deux groupes de phrasèmes : les *pragmatèmes* et les *phrasèmes sémantiques*. Les pragmatèmes sont :

— Les expressions dont le signifié et le signifiant ne sont pas construits sans restriction, même si elles sont construites d'une manière régulière, par ex. : *all you can eat*, *à tout à l'heure* ;

— Les expressions où seulement le signifié n'est pas construit sans restriction. Par exemple les salutations, les clichés techniques et les phrases comme *il est défendu de fumer*, etc.

Dans les phrasèmes sémantiques, le signifié est libre (il est construit sans restriction) et son signifiant n'est pas libre. Il y a trois types de phrasèmes sémantiques :

— L'expression idiomatique : le sens de l'expression est plus ample que les sens de ses mots constituants et ceux-ci ne sont pas inclus dans le sens de l'expression. Par exemple : *donner sa langue au chat* ;

— Le quasi-phrasème (ou expression quasi idiomatique) : le signifié de l'expression contient le signifié de ses mots constituants. Cependant, il contient aussi un signifié qui dépasse les signifiés des mots individuels, par ex. : *start a family*, *bed and breakfast*, etc. ;

— La collocation : le sens de l'expression inclut le sens de ses mots constituants. Un des constituants (m_1) est choisi librement et l'autre mot, ou expression, (m_2) est choisi en fonction de m_1. Nous pouvons avoir comme collocations (Manning et Schütze, 1999), par exemple, les constructions à verbe support : *prendre une décision*, *faire attention*, etc.

Dans cette thèse, nous considérons comme expression polylexicale tout type de phrasème et nous nous concentrons sur le troisième type de phrasème sémantique ici présenté : la *collocation*. Les collocations forment la majorité des phrasèmes et posent des difficultés à plusieurs tâches liées au TALN (Mel'čuk, 1998).

Selon Polguère (2000), les collocations sont des expressions semi-idiomatiques, ayant la forme L_1+L_2, où un des composants, le *collocatif*, est choisi pour exprimer un sens spécifique, dans un rôle syntaxique spécifique, contingent aux choix de l'autre composant, la *base*. L_2, le collocatif, peut être un lexème ou un phrasème.

Le choix du collocatif dépend fortement du lexème choisi comme base (Heid et Raab, 1989). Pour Hausmann (1979), les collocations sont des expressions phraséologiques compositionnelles.

Par exemple, dans la collocation *peur bleue*, *bleue* est le collocatif et *peur* est la base. Ici, le collocatif *bleue* est choisi pour exprimer un sens spécifique, le sens d'intensification de la lexie *peur* (base). Étant donné la base, il n'y a pas plusieurs mots (souvent, il n'y en a qu'un) qui expriment le sens spécifique qu'on désire exprimer (ici, le sens d'intensification).

En résumé, étant donnée la base *peur*, pour exprimer le sens spécifique *intensification*, le choix du collocatif qui exprime ce sens contingent à cette base est restreint à peu d'options : *bleue*, donc : intensification (*peur*) = {*bleue*}.

Dans la littérature, plusieurs travaux existent sur l'*identification* et l'*extraction* de collocations (et d'EPL en général) à partir de textes. Cependant, très peu se concentrent sur la *représentation* de collocations en format électronique (par ex. pour la construction d'ontologies, de dictionnaires ou de réseaux lexicaux) et sur l'utilisation de collocations dans les différentes tâches liées au TALN.

1.10 Conclusion

Dans ce chapitre, nous avons présenté les fondements théoriques liés à la représentation des connaissances, en particulier les connaissances linguistique et sémantique. Nous avons fait aussi le lien entre la catégorisation, la langue et la représentation ontologique.

Au chapitre suivant, nous allons présenter les principaux courants liés à la représentation linguistique, en soulignant les principes de la linguistique cognitive.

CHAPITRE II

THÉORIES LINGUISTIQUES ET REPRÉSENTATION LEXICALE

2.1 Introduction

La lexicologie est la branche de la linguistique qui étudie le lexique d'une langue (Eluerd, 2000). Saussure *et al.* (1972) l'ont définie comme la science des mots. Pour leur part, Lehmann et Martin-Berthet (1998) la définissent comme « la tâche d'établir la liste des unités qui constituent le lexique et de décrire des relations entre ces unités ».

La lexicologie se distingue de la lexicographie, dans le sens que cette dernière s'occupe de la construction des dictionnaires. La lexicographie étudie le lexique au niveau concret et son but est la production des dictionnaires pratiques. Elle se préoccupe aussi, par exemple, des problèmes typographiques et commerciaux (Mel'čuk *et al.*, 1995).

Pour sa part, la lexicologie étudie le lexique au niveau formel et abstrait. Son but est d'en dégager des lois générales et d'en proposer une formalisation. La lexicologie doit utiliser ce type de formalisation pour la description de l'ensemble des lexies. Son but est aussi la production d'un dictionnaire ; non pas un dictionnaire physique, mais « un dictionnaire idéalisé — le prototype des dictionnaires pratiques » (Mel'čuk *et al.*, 1995, p. 27).

Selon Fuchs et Le Goffic (1992), il y a trois courants linguistiques principaux pour expliquer la langue et les relations lexicales : le *courant structuraliste*, le *courant formaliste* (grammaire formelle ou générative) et les *théories sémantiques de l'activité du langage*. Nous divisons le troisième courant en deux autres : le *fonctionnalisme* et la *linguistique cognitive*.

À la section suivante, nous décrivons brièvement les premiers courants cités ci-dessus. À la section 2.3 nous présentons plus en détail la linguistique cognitive.

2.2 Courants linguistiques

Dans cette section, nous présentons brièvement les principaux courants linguistiques : le structuralisme, le formalisme et le fonctionnalisme.

2.2.1 Structuralisme

Le courant structuraliste se concentre sur la description de la structure de la langue. Par exemple, ce que Lyons (1977) appelle « structuralisme », c'est le structuralisme de Saussure.

Selon Saussure *et al.* (1972), chaque langue est un système ou une structure relationnelle. Les composants de cette structure sont les sons (phonèmes), les mots (lexèmes), les sens, etc. L'essence de ces composants vient de la relation qu'ils ont les uns avec les autres. Selon Lyons (1977) :

> Linguistic units are but points in a system, or network, of relations ; they are the terminals of these relations, and they have no prior and independent existence (Lyons, 1977, p. 232).

Parmi les exemples théoriques qui s'inscrivent dans le courant structuraliste, citons : la syntaxe structurale (Tesnière, 1959) et la sémantique componentielle (Pottier, 1992).

2.2.2 Formalisme

Le courant formaliste est basé sur les grammaires formelles ou génératives, initialement développées par Chomsky (1957).

Une grammaire formelle définit une série des règles de générations des chaînes de caractères, en suivant la syntaxe d'un langage. Seules les séquences de caractères (par exemple, des mots) pouvant être générées par les règles de la grammaire d'un langage appartiennent à ce langage.

2.2.3 Fonctionnalisme

Le fonctionnalisme est apparu avec l'École de Prague (Vachek, 1966). Il y a différentes théories considérées comme fonctionnalistes et elles ont en commun de faire porter l'accent sur l'usage de la langue pour comprendre son fonctionnement.

Par exemple, la langue a une fonction sémantique, celle de transmettre un sens en utilisant différents agents (actants sémantiques), tels que l'agent et le patient d'une action. Elle a aussi une fonction pragmatique : le thème est le sujet du discours (ou « de quoi on parle ») et le rhème est l'information nouvelle que le locuteur donne sur le thème.

Finalement, la linguistique cognitive est présentée plus en détail à la section suivante.

2.3 Linguistique cognitive

La linguistique cognitive est un ensemble de théories qui traitent les relations entre la langue, le cerveau et l'expérience socio-physique (Evans *et al.*, 2007).

Son origine est la philosophie, mais elle a également reçu l'influence d'autres sciences cognitives, comme la psychologie cognitive, la psychologie de la Gestalt

et les idées fondamentales de la langue et de la cognition. Il y a dans cet ensemble de théories deux points principaux : l'*engagement de généralisation* (*The Generalization Commitment*) et l'*engagement cognitif* (*The Cognitive Commitement*) (Lakoff, 1991; Evans *et al.*, 2007), que nous présentons ici. [1]

a) L'engagement de généralisation

Ce point représente les principes généraux applicables à tous les aspects des langues. À la différence de la linguistique formelle, qui sépare l'étude de la langue en différents domaines, comme la phonologie, la sémantique, la syntaxe, la pragmatique et la morphologie, la linguistique cognitive cherche à identifier les principes généraux des langues et la façon de les combiner pour produire les différents aspects des langues.

Par exemple, la linguistique formelle, représentée par la théorie de la grammaire générative (Chomsky, 2002) et par la théorie de la sémantique formelle (Partee, 1976), conçoit le cerveau de manière modulaire. Elle défend que différents domaines, comme la sémantique ou la syntaxe, utilisent différents types de primitives et que pour cette raison, ils devraient être expliqués par différents principes structurels.

Par contre, les représentants de la linguistique cognitive défendent que, malgré son utilité, cette approche modulaire ne devrait pas être le point de départ de la compréhension du fonctionnement des langues. Ils veulent comprendre les points en commun entre certains aspects des langues pour appliquer la connaissance obtenue à d'autres aspects. Par exemple, la compréhension des structures syntaxiques ou morphologiques peut être utile dans l'étude de la sémantique.

---

1. La continuation de cette section est un résumé du chapitre 2 de l'article d'Evans *et al.* (2007)

Un autre aspect important de la linguistique cognitive est sa vision « verticale » de la langue, qui est représentée comme étant formée par des couches successives ; la sémantique étant celle du dessus. À partir de cette vision verticale, on peut contempler les différents aspects en même temps, étudier des parties de chaque couche simultanément.

b) L'engagement cognitif

Selon les découvertes et les études les plus récentes sur le cerveau et sur l'esprit humain, faites par les sciences cognitives, comme l'intelligence artificielle, les neurosciences, la philosophie et la psychologie, l'engagement cognitif est l'engagement qu'il faut prendre envers la compréhension et la caractérisation des principes généraux de la langue.

En conséquence de cet engagement, certains chercheurs défendent que les théories linguistiques ne devraient pas prévoir ou établir des structures ou des processus qui ne s'accordent pas aux propriétés ou aux caractéristiques déjà connues sur le cerveau humain.

Une autre conséquence de cet engagement est que les modèles qui utilisent des propriétés cognitives humaines déjà observées sont préférables aux modèles basés sur des métriques du cerveau jamais observées empiriquement.

Dans les deux sous-sections suivantes, nous présentons les principes et les théories portant sur la *sémantique cognitive*, un des domaines de recherche de la linguistique cognitive.

2.3.1 Principes de la sémantique cognitive

Cette sous-section est un résumé des idées présentées au chapitre 4 de l'article d'Evans *et al.* (2007).

La sémantique cognitive est un des domaines de recherche de la linguistique cognitive. Il s'agit de l'investigation des relations entre l'expérience, le système conceptuel et la structure sémantique de la langue (Evans *et al.*, 2007).

Evans *et al.* (2007) identifient quatre principes de la sémantique cognitive :

a) La structure conceptuelle est incarnée

La construction de la réalité et des concepts qui sont autour de nous dépend de notre architecture neurologique et de notre anatomie (affordances ou potentialités). Par exemple, la manière dont les êtres humains voient les couleurs est différente de la manière dont les autres animaux les voient, et cela affecte notre classification des objets par rapport aux couleurs.

Les concepts et la nature de la réalité que nous pouvons concevoir sont reliés à l'architecture de notre corps et de notre cerveau. Les choses que nous pouvons comprendre et pour lesquelles nous créons des concepts dépendent de notre expérience et de notre contact avec la réalité du monde.

b) La structure sémantique est équivalente à la structure conceptuelle

La langue (ou les mots d'une langue) fait référence à des concepts qui sont représentés dans l'esprit, au lieu de faire directement référence à des entités qui sont externes. Dit autrement, la structure sémantique (le réseau sémantique associé aux mots) est équivalente à la structure conceptuelle représentée à l'interne.

Cependant, cela ne signifie pas que les deux sont la même chose. Les sens associés aux unités linguistiques forment un sous-ensemble des concepts possibles, car la quantité de pensées, de sentiments, de désirs et d'idées est beaucoup plus large que le lexique d'une langue. Il y a des concepts qui ne sont pas lexicalisés.

c) La représentation des connaissances est encyclopédique

Ça n'est pas le mot qui porte un sens ou une certaine quantité de sens en lui-même. Au contraire, le mot est un point d'accès à une connaissance encyclopédique, reliée à un concept ou domaine, et cette connaissance varie en fonction du contexte où le mot est inséré.

d) La construction de la connaissance est équivalente à la conceptualisation

La langue elle-même ne code pas le sens, mais elle incite la construction du sens, ce qui est construit au niveau conceptuel. Le sens est vu comme un processus, déclenché par la langue, au lieu d'être une chose statique, encodé par la langue. Une fois de plus le contexte est vu comme essentiel pour la formation du sens.

2.3.2 Théories à propos de la sémantique cognitive

Evans *et al.* (2007) présentent huit différentes théories à propos de la sémantique cognitive : le schéma d'images, la sémantique encyclopédique, le modèle cognitif idéalisé, la sémantique lexicale cognitive, la métaphore conceptuelle, la métonymie conceptuelle, les espaces mentaux et le mélange conceptuel. Cette sous-section est un résumé des idées présentées au chapitre 5 de l'article d'Evans *et al.* (2007).

a) Schéma d'images

Selon cette théorie, qui a d'abord été proposée par Johnson (1990) et développée par Lakoff (1991), une des formes de manifestation de nos expériences incarnées, au niveau cognitif, est le schéma d'images.

Ces schémas sont formés par nos expériences linguistiques, nos interactions avec le monde et par le contexte historique. Les expériences et les interactions forment des pré-concepts, qui ne sont pas des abstractions désincarnées : ils sont dérivés du contact de nos sens corporels avec le monde.

Ces pré-concepts donnent lieu à des relations qu'on appelle métaphores concep-

tuelles. Par exemple, le concept de « récipient », comme un espace fermée, qui est appris quand on est encore un enfant, est utilisé métaphoriquement en différentes langues et dans d'autres situations. Par exemple, dans « dans ce moment-là », le mot *dans* ne signifie pas physiquement à l'intérieur d'une chose ou place, mais il possède un sens figuratif, comme « à l'intérieur d'un espace de temps ». Autre exemple : dans « je ressens de la pression pour finir mon projet », le concept physique de *pression* est utilisé métaphoriquement pour donner l'image d'une force extérieure qui nous pousse à faire quelque chose.

De cette manière, quelques concepts, plus abstraits, peuvent être créés à partir d'autres, plus concrets.

b) Sémantique encyclopédique

Selon cette théorie, ce n'est pas le mot qui porte le sens : le sens d'un mot est diffusé parmi les différents textes et conversations et il varie selon le moment et le contexte. En conséquence, il n'y a pas de différence entre la sémantique et la pragmatique. Le mot fonctionne comme un point d'accès à la connaissance qu'il représente.

Cependant, cela ne veut pas dire que la connaissance encyclopédique est désorganisée. Elle est dynamique et s'organise comme un réseau. Être dynamique signifie qu'elle change avec le temps. Par exemple, notre compréhension du mot « *chapeau* » augmente si on visite un pays étranger où il y a des chapeaux que nous ne connaissions pas avant.

c) Modèle cognitif idéalisé

Selon Lakoff (1987), la connaissance représentée par un cadre sémantique, comme FrameNet, est normalement une conceptualisation de l'expérience qui ne correspond pas complètement à la réalité.

Cela signifie que les modèles cognitifs sont des idéalisations : ils ne peuvent pas représenter la réalité avec 100% de fidélité.

En conséquence, peu importe le modèle, son degré de précision et de granularité ; il y aura toujours certains éléments d'une terminologie, d'une taxonomie ou d'une ontologie qui se situeront à la frontière entre deux ou plusieurs catégories ou termes.

Par exemple, si nous avons la catégorie *vêtement* dans une ontologie, doit-on considérer *chaussure* comme un membre de cette catégorie ?

On doit aussi considérer l'effet de typicité. Selon l'approche prototypique (Rosch, 1975), un prototype représente un cas moyen de tous les individus d'une catégorie. Par exemple, il y a différents types de chats, ayant différentes couleurs, tailles, types de fourrure, etc. Mais un chat de la race des *sphinx* est plus loin du *chat* prototype ; il est un représentant moins typique. Par contre, il y a d'autres représentants qui sont plus typiques, qui sont de « bons » exemplaires de la catégorie. Par exemple, un chat de la race des *siamois* est un chat prototype.

d) Sémantique lexicale cognitive

Selon la sémantique lexicale cognitive, les mots sont des catégories conceptuelles. Un mot représente une catégorie qui contient des sens distincts, connectés les uns aux autres. L'effet de typicité s'applique aux différents sens d'un mot : il y a des sens qui sont plus typiques et d'autres qui sont moins typiques.

Par exemple, le mot français *puce* a plusieurs sens (un insecte, une puce électronique, une couleur (marron), etc.). Ces sens sont liés les uns aux autres de différentes manières, mais il y a un sens (*insecte*) qui est plus typique que les autres.

e) Métaphore conceptuelle

Cette théorie défend l'idée que la pensée est métaphorique. La structure de la connaissance est organisée en relations de domaines qui s'entrelacent. Certaines de ces relations sont formées par des expériences pré-conceptuelles incarnées et d'autres sont construites sur ces premières expériences pour former des structures conceptuelles plus complexes.

Par exemple, dans la phrase : « il a un potentiel très élevé », l'idée de qualité, de compétence, est exprimée en utilisant un concept de verticalité. Le niveau le plus haut correspond au meilleur, au plus désirable et le niveau le plus bas correspond à une qualité ou à un potentiel moindre. L'idée physique de verticalité est incarnée dans le monde externe et l'idée métaphorique de verticalité est construite sur l'idée physique.

f) Métonymie conceptuelle

La *métonymie* est une figure de style qui correspond au remplacement d'un mot (ou expression) par un autre mot (ou expression) connecté logiquement à celui remplacé. Par exemple, on peut dire : « l'industrie est en grève » au lieu de : « les travailleurs de l'industrie sont en grève ».

Tandis que dans la métaphore il y a une connexion logique par analogie, dans la métonymie, la connexion s'établit par contiguïté.

Pour quelques théoriciens de la sémantique cognitive (Lemmens, 2015), la métonymie est plus qu'une figure de style : elle est responsable de créer de nouveaux concepts. Certains chercheurs (*Ibid.*) la considèrent comme la base de la métaphore conceptuelle.

Par exemple : les immigrants illégaux en France n'ont pas de documents d'iden-

tification (des papiers) et sont connus comme des *sans-papiers*, au point où l'expression *sans-papier* devient un nouveau concept, un synonyme pour désigner l'*immigrant illégal*.

g) Espaces mentaux

Selon Fauconnier et Turner (2002), la construction du sens comporte deux processus : la construction des *espaces mentaux* et la création des relations entre eux.

Les espaces mentaux sont des régions de l'espace conceptuel et chacun contient ses types spécifiques d'information (Evans *et al.*, 2007). Ils sont construits à partir de notre interaction avec le monde, les personnes, les différentes situations. Chaque espace est temporaire : il sont créés et utilisés sous une forme pragmatique et dépendante de leur contexte présent. Les processus de formation des espaces mentaux et les relations entre eux offrent la possibilité de créer différents sens pour différents contextes.

Selon cette théorie, chaque mot ou expression n'a pas un sens déterminé, mais plutôt un sens potentiel, qui change en fonction du contexte où ils sont utilisés.

h) Mélange conceptuel

Le point principal de la théorie du mélange conceptuel est que le sens d'un mot ou d'une expression est formé par ses parties constituantes, qui font surgir une structure émergente, dont le sens est plus vaste que la somme des sens des parties.

Il y a un point en commun entre cette théorie et la théorie des espaces mentaux, par rapport à la vision dynamique des sens des unités linguistiques.

C'est à partir de cette synergie que l'imagination humaine est développée. Et l'imagination est un des acteurs les plus importants dans les processus cognitifs,

incluant la construction des sens, des concepts, des nouveaux termes, etc.

2.4 Conclusion

Dans ce chapitre, nous avons fait le résumé des différents courants linguistiques et de leurs visions par rapport à la représentation lexicale, conceptuelle et sémantique.

Nous avons vu plus en détail les idées proposées par la linguistique cognitive, spécialement les théories à propos de la sémantique cognitive. Certaines des idées de la sémantique cognitive sont dispersées dans les sujets que nous abordons dans cette thèse. Par exemple :

— La théorie du *schéma d'image* réfère à des sens métaphoriques qui sont dérivés des sens concrets. Dans notre modèle, par exemple, le vocable *OCÉAN* est représenté comme une *entrée lexicale* et ses différents sens, comme *Océan*$_I$ (sens concret) et *Océan*$_{II}$ (sens abstrait, comme dans *océan de gens*), sont représentés comme des *sens lexicaux*. La théorie de la métaphore conceptuelle, ainsi que la théorie de la métonymie conceptuelle, aident aussi à expliquer l'émergence de sens abstraits à partir des sens concrets.

— Selon la *sémantique encyclopédique*, le sens d'un mot est diffusé par les différentes façons dont il est utilisé et il dépend de ses relations avec d'autres mots. Il s'agit d'une des motivations pour la construction des réseaux lexicaux, présentés au chapitre 5. La théorie des *espaces mentaux* est aussi pertinente pour la conceptualisation des réseaux lexicaux/sémantiques.

— La théorie du *mélange conceptuel*, qui affirme que le sens d'une expression est plus large que la combinaison des sens de ses parties constituantes, est importante pour la compréhension de l'idée de collocation, que nous avons présentée à la section 1.9.

Au chapitre 3, nous présentons la théorie linguistique Sens-Texte (Žolkovskij et Mel'čuk, 1967; Mel'čuk, 1997) et au chapitre 4 les fonctions lexicales (Mel'čuk, 1995), un outil développé dans le cadre de la théorie Sens-Texte pour la représentation de relations lexicales/sémantiques.

CHAPITRE III

THÉORIE SENS-TEXTE

3.1 Introduction

La théorie Sens-Texte (TST) est une théorie linguistique qui présente des caractéristiques des courants structuralistes et fonctionnalistes (Tremblay, 2009). Son origine remonte aux années 1960 (Žolkovskij et Mel'čuk, 1965). Selon Polguère (1998), la TST peut être décrite par les cinq propositions suivantes :

P1) « La TST rend compte de l'association que tout locuteur d'une langue L est capable de faire entre un sens donné de L et l'ensemble des énoncés paraphrastiques de L exprimant ce sens » ;

P2) « La TST est universelle, c'est-à-dire qu'elle repose sur des principes généraux s'appliquant à toutes les langues » ;

P3) « La TST est linguistique, en ce sens qu'elle permet, à partir des principes généraux sur lesquels elle repose, de construire des modèles linguistiques spécifiques pour chaque langue humaine » ;

P4) « La TST permet de construire des modèles calculables » ;

P5) « La TST est formelle ».

Le point principal de cette théorie est que le locuteur a, dans son esprit, des

intentions (désirs, croyances, etc.) à exprimer. Cependant, il y a plusieurs manières d'exprimer le sens de son intention, chacune étant un « énoncé paraphrasique ».

Par exemple, si on veut exprimer la croyance que « l'armée a résisté à l'attaque de l'ennemi », on pourrait dire :

a) « L'armée a offert une résistance à l'ennemi » ou ;

b) « L'armée a résisté à l'ennemi ».

Ces deux manières d'exprimer un même sens sont deux différents « énoncés paraphrasiques ». Un des objectifs principaux de la TST a été, dès son origine, de systématiser les règles de paraphrases, ou, comme le dit Mel'čuk (1992) : « Nos propositions visaient la SYNTHÈSE AUTOMATIQUE DE PHRASES PAR ORDINATEUR - à partir d'une représentation sémantique d'une famille de phrases (plus ou moins) synonymes » (*Ibid.*, p. 9).

Par exemple, Mel'čuk montre qu'en français il y a, en théorie, 4 374 000 paraphrases de la phrase suivante :

« Il est clair que la manière dont la police en U.R.S.S. persécute les gens de lettres est aujourd'hui très différente de ce qu'elle était il y a cinq décennies. »

Cet exemple montre la raison du nom « Sens-Texte ». Il y a, dans une langue, une quantité infinie de sens, une quantité infinie de textes (formes d'expression) et plusieurs correspondances entre sens et textes (Mel'čuk, 1997). Par exemple, une croyance, une information qu'un locuteur désire exprimer (le sens) peut être exprimée de plusieurs manières et un texte (phrase, mot, etc.) peut avoir plusieurs sens (polysémie).

Plus formellement, Polguère propose que

> [l]a langue, selon la TST, est un système de règles lexicales et grammaticales qui, appliquées de façon séquentielle du niveau sémantique jusqu'au niveau phonétique et vice versa, établissent une correspondance bidirectionnelle Sens , Texte (Polguère, 2011, p. 5).

Dans la TST, on priorise la direction du sens vers le texte (parler, faire la synthèse). Le contraire, le passage du texte vers le sens (comprendre la parole, faire l'analyse), l'interprétation du texte, n'est pas le sujet de l'étude de la TST. Nous verrons à la section 3.3 comment la TST présente le passage du sens vers le texte. Au chapitre 4, nous montrons comment les fonctions lexicales sont utilisées pour la systématisation de paraphrases et à la section 4.8, nous parlons des règles de paraphrases de la TST.

Le fait que la TST insiste sur la fonction de communication de la langue est la raison pour laquelle elle est considérée comme une théorie fonctionnaliste (Tremblay, 2009).

Les propositions P2 et P3 indiquent que la TST est universelle. Ses principes sont observables et applicables à pratiquement toutes les langues.

Selon les propositions P4 et P5, à partir de la TST, il est possible de construire des modèles calculables, c'est-à-dire des modèles qui sont traitables par une machine de Turing (Turing, 1936).

Turing (1995) a affirmé qu'une fonction est effectivement calculable si ses valeurs peuvent être trouvées d'une façon purement mécanique (faite par une machine). Comme les processus mécaniques sont habituellement reconnus comme des paradigmes pour des processus constructifs et finis, la notion courante d'une procédure effective est devenue celle d'une procédure qui peut être réalisée par une machine de Turing.

Les symboles d'une machine de Turing sont formels. Cela signifie que peu importe quels sont ces symboles, l'important est qu'on puisse les distinguer les uns des autres par une ou plusieurs propriétés physiques. Ainsi, dans différentes réalisations de la même machine de Turing, le même symbole peut être instancié par différents objets.

De son côté, la TST est aussi formelle, au sens où elle établit des formalismes pour représenter des relations entre sens (comme le réseau sémantique), des relations lexicales (les fonctions lexicales) et des règles de passage d'une représentation à une autre (par exemple, le passage du réseau sémantique à l'arbre syntaxique profond).

Les propositions P4 et P5, et le fait que la TST utilise des notions de la syntaxe structurale de Tesnière (1959) comme la dépendance syntaxique, montrent pourquoi on considère la TST aussi comme une théorie structuraliste.

En ce qui concerne les principes de la linguistique cognitive, la TST ne s'intéresse pas au fonctionnement du cerveau et n'essaie pas d'expliquer comment le sens est représenté dans le cerveau.

La TST est la première théorie linguistique à identifier et à formaliser les relations paradigmatiques et syntagmatiques au niveau des lexies, en créant les foncions lexicales.

3.2 Notions importantes de la théorie Sens-Texte

Nous présentons dans cette section les notions de la TST qui sont fondamentales pour la compréhension des fonctions lexicales et de notre modèle ontologique de représentation des fonctions lexicales.

3.2.1 Actant sémantique

En logique de premier ordre, un prédicat est une relation entre des entités ou des objets, et cette relation peut retourner les valeurs « vrai » ou « faux ». Par exemple, on peut définir le prédicat binaire capitale(X,Y) et écrire les faits : capitale(*Canada*, *Ottawa*), capitale(*France*, *Paris*).

Selon Mel'čuk *et al.* (1995), « les prédicats sémantiques désignent des actions, des événements, des processus, des états, des propriétés, des relations, etc. — en un mot, des faits qui impliquent nécessairement des participants ». Dans l'exemple ci-dessus, *Canada* et *Ottawa* sont les arguments (ou participants) du prédicat *capitale*, qui est d'arité 2.

En linguistique, le prédicat reçoit le nom *sens prédicatif* et les arguments sont les *actants sémantiques* (*ASem*). Le sens *donner*, par exemple, est un sens prédicatif avec trois actants sémantiques :

1. Quelqu'un donne 2. quelque chose à 3. quelqu'un d'autre (Mel'čuk *et al.*, 1995).

Dans le cas de « donner » :

- l'$ASem_1$ est l'exécuteur de l'action (ou événement, processus, etc.) représentée par le sens prédicatif ;
- l'$ASem_2$ est l'entité qui fonctionne comme l'objet de l'action ;
- l'$ASem_3$ est l'entité qui reçoit ou qui subit l'action ;
- l'$ASem_{xt}$ représente un actant externe à l'action.

3.2.2 Actant syntaxique profond

Un actant syntaxique profond (*ASyntP*) d'une lexie *L* est « un syntagme qui dépend de *L* syntaxiquement et en exprime un actant sémantique » (Mel'čuk

et al., 1995).

En voici quelques exemples :

— « Jean est arrivé » : *Jean* est l'ASyntP$_I$ de la lexie *arrivé* ;

— « échec militaire » : *militaire* est l'ASyntP$_I$ de la lexie *échec*, même s'il n'est pas explicitement le sujet de la phrase, car l'action de *échouer* est « exécuté » par *militaire* ;

— « Jean a reçu un conseil de Pierre » : *Jean* est l'ASyntP$_{III}$ de *conseil*, même si *Jean* est le sujet grammatical (SG) de la phrase, car l'idée est que « X conseille Y à Z », et *Pierre* est l'ASyntP$_I$ de la lexie *conseil* (celui qui donne le conseil, qui exécute l'action).

Pour les exemples précédents, nous avons :

— L'ASyntP$_I$ d'une lexie verbale *L* correspond à l'exécuteur de l'action (processus, etc.) exprimée par *L* ;

— L'ASyntP$_{II}$ de *L* est l'objet de l'action exprimée par *L* ;

— L'ASyntP$_{III}$ est le destinataire de l'action exprimée par *L*.

3.2.3 Actant syntaxique de surface

Les actants syntaxiques de surface (ASyntS) correspondent au sujet, à l'objet direct et à l'objet indirect de la grammaire traditionnelle (Mel'čuk *et al.*, 1995).

Il faut noter qu'un ASyntS$_I$ ne correspond pas nécessairement à un ASyntP$_I$. Un ASyntS$_I$ peut correspondre à un ASyntP$_{II}$, ou encore, un ASyntS$_{II}$ peut correspond à ASyntP$_I$. Par exemple, considérons la phrase suivante :

« Jean a reçu un conseil de Pierre. »

Jean est l'ASyntS$_I$ de cette phrase, car il est en le sujet. La lexie *conseil* est l'ASyntS$_{II}$, car il est l'objet direct. *Pierre* est l'ASyntS$_{III}$, car il est l'objet indirect.

Cependant, tel qu'expliqué à la sous-section précédente, *Jean* est l'ASyntP$_{III}$, car c'est lui qui reçoit l'action et *Pierre* est le l'ASyntP$_I$, car c'est lui qui exécute l'action.

3.2.4 Dépendance syntaxique

L'idée de *dépendance syntaxique* tire son origine des travaux de Tesnière (1959). Pour chaque mot m_1 dans une phrase, il n'y a qu'un autre mot m_2 qui dépend syntaxiquement de m_1.

Dans une analyse syntaxique de dépendance, le verbe est la racine de la structure de la phrase et il y a toujours des relations binaires entre un nœud et les nœuds qui sont plus bas dans l'arbre syntaxique. Contrairement à la constituance syntaxique, il n'existe pas pour la dépendance syntaxique de division initiale d'une clause en phrase nominale (le sujet) et phrase verbale (le prédicat).

En comparaison, dans la *constituance syntaxique*, la structure d'une phrase est formée par des relations constitutives entre un sujet (phrase nominale) et un prédicat (phrase verbale). Le sujet est formé, par exemple, par un déterminant et un nom, le prédicat par un verbe et une deuxième phrase verbale. La deuxième phrase verbale est formée par d'autres parties, etc. Il y a toujours une division binaire de la phrase (et des sous-phrases), et pour cette raison, il y a toujours des relations de type *un-plusieurs* (à l'exception des relations vers les feuilles) dans l'arbre syntaxique qui représente la phrase. Chaque mot dans une phrase peut dépendre syntaxiquement d'un ou de plusieurs mots. Cela fait qu'un arbre de constituance syntaxique est moins compact qu'un arbre de dépendance syntaxique.

Une grammaire de dépendance est basée sur la dépendance syntaxique, alors

qu'une grammaire de constituants est basée sur la constituance syntaxique.

3.2.5 Représentation des différents sens d'un mot

Dans la TST, les différents sens d'un mot sont représentés par des indices, en utilisant des chiffres romains et arabes et des lettres latines (Mel'čuk *et al.*, 1995), ce que nous illustrons avec un exemple :

Considérons les définitions du mot *océan*. Il a des sens concrets, par exemple « une extension d'eau qui couvre la planète », et des sens abstraits, comme dans « l'océan de gens ». Dans la TST, les sens concrets d'océan peuvent être représentés comme Océan$_{I}$ et les sens abstraits comme Océan$_{II}$. Dans Océan$_{I}$, il peut y avoir des sous-divisions :

— *Océan*$_{I:1a}$: « une extension d'eau qui couvre la planète » (toujours au singulier) ;

— *Océan*$_{I:1b}$: « l'ensemble des océans en général » (toujours au pluriel), comme dans « les océans sont pollués »

— *Océan*$_{I:2}$: « une division de *Océan*$_{I:1a}$ dans une région spécifique », par exemple : *Océan Atlantique*, *Océan Pacifique*, *Océan Arctique*, etc.

3.3 Niveaux de modélisation structurale de la théorie Sens-Texte

Il y a sept niveaux de modélisation structurale dans la TST (Žolkovskij et Mel'čuk, 1967; Polguère, 2011) :

i) Représentation sémantique : graphe sémantique dont les arcs sont des relations prédicat-argument entre sens ;

ii) Représentation syntaxique profonde : arbre des dépendances syntaxiques universelles, dont les nœuds sont des lexies pleines ou des locutions (par

exemple, des collocations). Les collocatifs apparaissent comme une application des fonctions lexicales ;

iii) Représentation syntaxique de surface : arbre des dépendances syntaxiques propres à chaque langue ;

iv) Représentation morphologique profonde : chaîne (linéaire) de lexies accompagnée de caractéristiques flexionnelles ;

v) Représentation morphologique de surface : chaîne de morphèmes qui forment la phrase ;

vi) Représentation phonologique profonde : chaîne de phonèmes ;

vii) Représentation phonologique de surface : chaîne de phones.

Cette structuration par niveaux permet le passage régulier du niveau sémantique au niveau phonologique de surface, de façon à ce que chaque interface entre niveaux adjacents soit harmonieuse, organisée selon des règles bien définies et pouvant être implémentée par une machine.

La figure 3.1 présente l'architecture de modélisation de la TST (Mel'čuk, 1997, p. 20).

Nous nous intéressons aux représentations sémantiques et syntaxiques et à l'interface sémantique-syntaxique liée aux fonctions lexicales. Par conséquent, nous présentons plus en détail les trois premiers niveaux dans les sous-sections suivantes.

Pour montrer ces trois niveaux de représentation, nous utilisons la phrase (1) comme exemple. Cette phrase et les figures qui suivnt sont extraites de Polguère (2011).

« Marc estime beaucoup Clara. » (1)

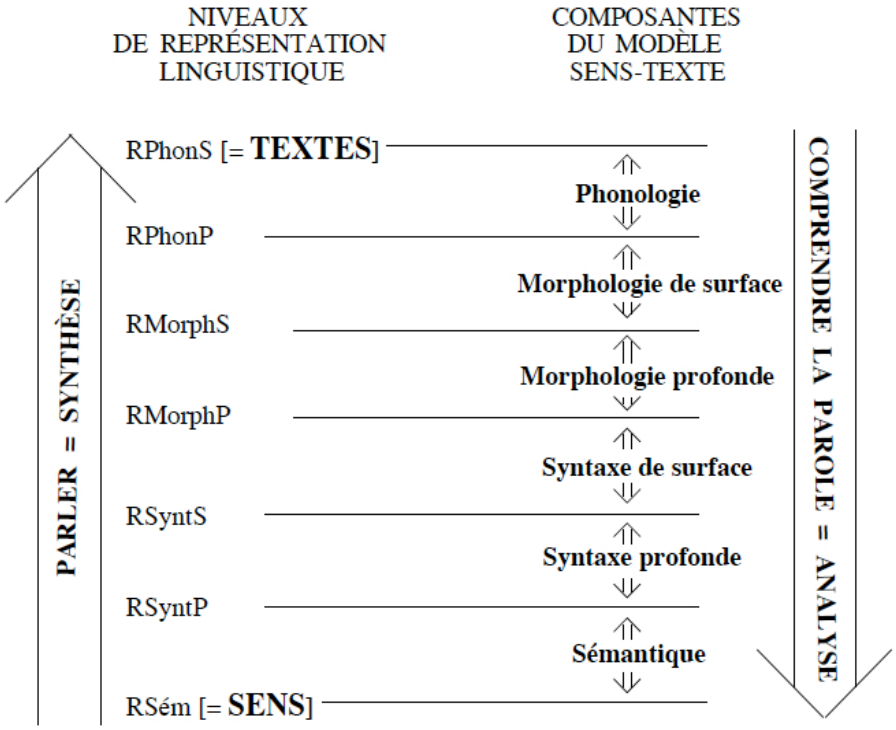

Figure 3.1: Les sept niveaux de modélisation de la théorie Sens-Texte (Polguère, 2011)

3.3.1 Représentation sémantique

La figure 3.2 montre la représentation sémantique de la phrase (1).

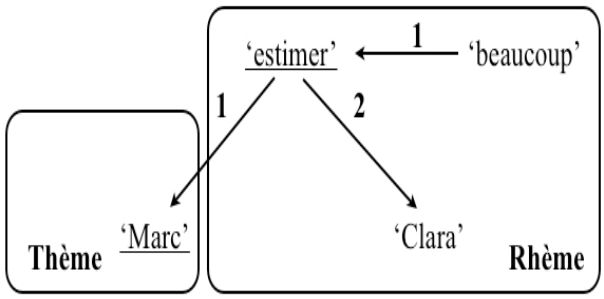

Figure 3.2: Représentation sémantique de la phrase « Marc estime beaucoup Clara. » (Polguère, 2011)

Premièrement, notons que la figure est divisée par thème et rhème. Le thème,

Marc, est le sujet du discours, ou « de quoi on parle ». Le rhème est l'information que le locuteur donne sur le thème, dans l'exemple, que *Marc* « estime beaucoup Clara ». Cette information pourrait être découpée d'une manière différente : par exemple, le thème pourrait être *Clara* et le rhème « est très estimée par Marc ».

Les flèches numérotées représentent les actants sémantiques : *Marc* est l'$ASem_1$ du verbe *estimer* et *Clara* est l'$ASem_2$. Ici, *beaucoup* représente une modification (intensification), qui est un prédicat sémantique d'un seul actant appliqué à *estimer*.

Il est important de noter que c'est une représentation sémantique. La phrase n'est pas représentée telle quelle, mais seulement par son sens : « quelqu'un estime quelqu'un d'autre d'une manière intense ». Il y a plusieurs paraphrases possibles pour exprimer ce sens : « Marc estime beaucoup Clara », « Marc a beaucoup d'estime pour Clara », « Marc tient Clara en haute estime », etc.

3.3.2 Représentation syntaxique profonde

La figure 3.3 montre la *représentation syntaxique profonde* de la phrase (1).

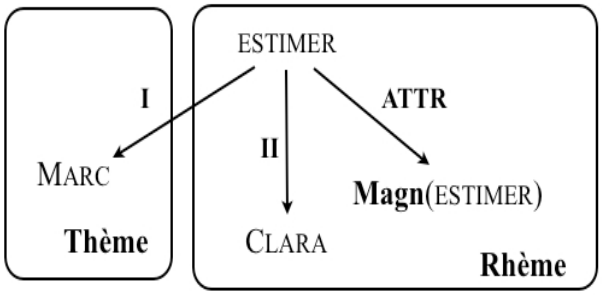

Figure 3.3: Représentation syntaxique profonde de la phrase « Marc estime beaucoup Clara. » (Polguère, 2011)

Ici, les flèches numérotées par des chiffres romains représentent des actants syntaxiques profonds. *Marc* est l'$ASyntP_1$ du verbe *estimer* et *Clara* est l'$ASyntP_2$.

Le prédicat d'intensification, représenté par *beaucoup* dans le réseau sémantique, est ici représenté par la fonction lexicale *Magn*, qui est expliquée au chapitre 4. Pour le moment, il suffit de noter que *Magn*(*estime*) = {*beaucoup*} et que *Magn* représente le sens prédicatif *intensification*.

Le symbole *ATTR* (attribut) exprime que *Magn*(*estime*) est un dépendant attributif d'*estime*. Dans ce cas, c'est *estime* qui devrait être l'actant syntaxique de *Magn*(*estime*), mais comme *estime* est la tête de la relation, la direction de la relation est inversée et *Magn*(*estime*) est représenté comme un *ATTR*.

3.3.3 Représentation syntaxique de surface

La figure 3.4 montre la *représentation syntaxique de surface* de la phrase (1).

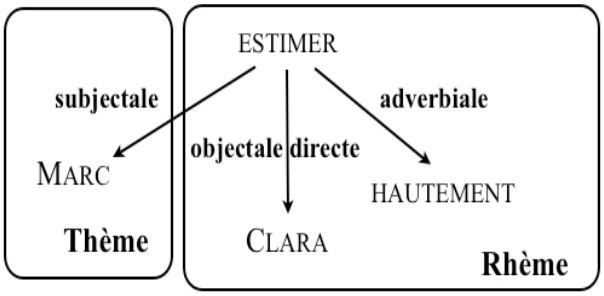

Figure 3.4: Représentation syntaxique de surface de la phrase « Marc estime beaucoup Clara. » (Polguère, 2011)

Dans la représentation syntaxique de surface se produit la lexicalisation du sens vers la langue cible (ici, le français). Les relations sont nommées comme dans la grammaire traditionnelle (sujet, objet direct, etc.) et la relation attributive est instanciée comme une relation adverbiale.

Nous montrons avec cet exemple un passage graduel et systématique de la représentation sémantique vers la représentation syntaxique de surface (spécifique d'une langue). Ce sont des caractéristiques qui rendent la TST modélisable et

implémentable d'un point de vue informatique.

3.4 Conclusion

Nous avons présenté dans ce chapitre les caractéristiques et définitions de la TST qui sont les plus pertinentes pour le développement de cette thèse.

Au chapitre suivant, nous présentons les fonctions lexicales, un outil développé dans le cadre de la TST pour représenter les différents types de relations lexicales.

Comme nous l'avons vu, un même sens peut être représenté par plusieurs paraphrases. À la section 4.8, après avoir introduit les fonctions lexicales, nous montrons, avec quelques exemples de règles de paraphrase, comment la TST systématise et catégorise le paraphrasage et comment les fonctions lexicales sont utilisées pour la modélisation de paraphrases.

CHAPITRE IV

FONCTIONS LEXICALES

4.1 Introduction

Les fonctions lexicales sont un formalisme pour la description et l'utilisation des propriétés combinatoires de lexèmes individuels (Bolshakov et Gelbukh, 1998). Wanner (2004) affirme que les fonctions lexicales peuvent être utilisées pour la description systématique des relations lexicales « institutionnalisées ». C'est un mécanisme puissant pour les transformations syntaxiques, la désambiguïsation lexicale dans les analyses sémantiques et le choix lexical lors de la génération de texte.

Les fonctions lexicales modélisent des relations sémantiques et combinatoires entre lexies (Polguère, 2000, p. 518). Ces relations ne sont pas morphologiques, car il n'y a pas nécessairement de relation entre les formes des lexies. Par exemple, la relation entre *chanter* et *chantent* est morphologique. Par contre, la relation entre *meurtre* et *victime* est sémantique.

Ce mécanisme est aussi utile pour l'analyse automatique de textes, la traduction automatique, le paraphrasage, etc. Les fonctions lexicales font partie de la théorie Sens-Texte (Mel'čuk, 1997; Valente, 2002) et constituent un outil important pour la modélisation des collocations, d'une manière systématique et exhaustive (Mel'čuk, 1998).

4.2 Définition d'une fonction lexicale

Une fonction lexicale (FL) est une fonction f qui associe à une lexie L (base, argument ou mot-clé de la fonction) un ensemble d'expressions lexicales $\{L_i\}$ plus ou moins synonymes (Mel'čuk, 1998) : $f\ (L) = \{L_i\}$. L'ensemble d'expressions est choisi en fonction de L.

On peut penser à une FL comme à un sens général et abstrait, qui peut être lexicalisé d'une manière concrète par les lexies qui expriment ce sens, par rapport à un argument donné. Les FL sont générales et indépendantes des langues.

Par exemple, la fonction *Magn* a le sens général et abstrait d'intensification, ou de *très*. Appliquée à l'argument *têtu*, elle produit comme résultat les expressions lexicales *comme un âne* et *comme une mule*, car en français, l'intensificateur de *têtu* (très têtu) s'exprime par les expressions *têtu comme un âne* et *têtu comme une mule*.

Polguère (2003) définit comme suit cette relation entre une FL et son sens abstrait :

> [...] les FL doivent avant tout être appréhendées comme des méta-lexies : elles sont caractérisées par des propriétés de sens et de combinatoire, mais ne sont pas associées de façon directe à des signes, et donc à des signifiants. L'association à des signifiants particuliers se fait, de façon indirecte, par l'application de la méta-lexie (en tant que fonction) à une lexie particulière, préalablement identifiée par le locuteur. Ces méta-lexies universelles devraient être enseignées comme telles. La FL *Magn* existe, même si c'est une notion « abstraite ». Elle existe au même titre que la notion de partie du discours, ou de genre grammatical (Polguère, 2003, p. 125-126).

Nous allons montrer ces définitions à l'aide des exemples suivants, dans lesquels ART signifie *article (déterminant)* (défini ou indéfini) et N signifie *nom*. Le sym-

bole « » doit être remplacé par la base, qui est le mot entre parenthèses.

Par exemple, lorsque la fonction *Oper*$_1$ est appliquée à une lexie (la base), elle retourne le verbe support qu'on utilise pour exprimer, dans la langue en question, le sens d'*e ectuer* ou d'*opérationnaliser*.

Les indices *1, 2, 3, etc.*, qui apparaissent dans le nom d'une fonction, sont des indices actanciels et représentent le premier, deuxième, troisième, etc., actant sémantique, tel qu'expliqué à la sous-section 4.6.1 :

— Oper$_1$ appliquée à *erreur* :

- Oper$_1$ (*erreur*) = {*faire* [ART]} (*faire une erreur*) ;

— Oper$_2$ appliquée à *ordre* :

- Oper$_2$ (*ordre*) = {*recevoir* [ART]} (*recevoir un ordre*).

Dans le cas d'*Oper*$_1$ (*erreur*), on peut aussi avoir comme résultat *commettre*. Cela signifie que *faire* et *commettre* sont « plus ou moins » synonymes par rapport à l'argument *erreur*.

Pour le dernier exemple, d'après le formalisme, on a :

— *f* (fonction) = *Oper*$_1$;

— *L* (base ou argument) = *erreur* ;

— {L_i} = {*faire; commettre*} ;

— *f(L)* = {L_i} ! Oper$_1$ (*erreur*) = {*faire, commettre*}.

À la section suivante, nous présentons une convention d'écriture des sous-scripts et exposants de certaines FL, qui utilisent une notation différente de celle utilisée dans la TST.

4.3 Conventions de notation des sous-scripts et exposants

Certains changements de notation ont été faits par rapport à la notation utilisée par la TST, et ce pour deux raisons :

1. Éviter l'utilisation des symboles qui ne sont pas correctement représentés dans certains systèmes informatiques, comme les systèmes de modélisation d'ontologies ;
2. Utiliser une nomenclature en anglais au lieu d'utiliser le français, en vue d'atteindre une audience plus globale.

Les conventions sont montrées en trois tableaux. Le tableau 4.1 rassemble les conventions concernant les FL spéciales, qui sont présentées à la section 4.5. Au tableau 4.2 se trouvent les conventions concernant la dimension de l'intensification des FL *Magn*, présentées à la sous-section 4.6.5. Finalement, le tableau 4.3 montre les conventions concernant les degrés d'équivalence des FL *Syn*, qui sont présentées à la sous-section 4.6.7.

Phénomène	Not. TST	Not. Thèse	Exemple
Changement d'actant	’	XYChange	Oper_1’ ! $\mathrm{Oper}_1{}^{\mathrm{XYChange}}$
Ensemble des individus	*f1g*	setOfInd	Oper_{f1g} ! $\mathrm{Oper}_1{}^{\mathrm{setOfInd}}$
Actant d'un actant	1[1]	11	$\mathit{Real}_{1[1]}$! Real_{11}

Tableau 4.1: Convention de notation des sous-scripts et exposants des FL spéciales

Au tableau 4.1, la première colonne montre le nom du phénomène linguistique représenté. La deuxième colonne montre le symbole utilisé par la TST. La troisième colonne montre le symbole utilisé dans cette thèse. Finalement, la quatrième colonne montre un exemple de FL où les types de notation sont utilisés. D'autres

exemples sont montrés aux sections 4.5 et 4.6.

Not. TST	Not. Thèse	Exemple
comportement	behaviour	$Magn_{comportement}$! $Magn_{behaviour}$
e□et	e□ect	$Magn_{effet}$! $Magn_{effect}$
épaisseur	thickness	$Magn_{paisseur}$! $Magn_{thickness}$
hauteur	height	$Magn_{hauteur}$! $Magn_{height}$
largeur	width	$Magn_{largeur}$! $Magn_{width}$
longueur	length	$Magn_{longueur}$! $Magn_{length}$
puissance	strength	$Magn_{puissance}$! $Magn_{strength}$
taille	size	$Magn_{taille}$! $Magn_{size}$
vitesse	speed	$Magn_{vitesse}$! $Magn_{speed}$

Tableau 4.2: Convention de notation des sous-scripts des FL Magn

Au tableau 4.2, la première colonne montre le symbole utilisé par la TST. La deuxième colonne montre le symbole utilisé dans cette thèse. La troisième colonne montre les FL *Magn* avec leurs sous-scripts respectifs. D'autres exemples sont montrés à la sous-section 4.6.5.

Not. TST	Not. Thèse	Exemple
∩	inter	$Syn_{\cap}$! Syn_{inter}
	more	Syn ! Syn_{more}
	less	Syn ! Syn_{less}

Tableau 4.3: Convention de notation des sous-scripts des FL Syn

Au tableau 4.3, la première colonne montre le symbole utilisé par la TST. La

deuxième colonne montre le symbole utilisé dans cette thèse. La troisième colonne montre les FL *Syn* avec leurs sous-scripts respectifs. D'autres exemples sont montrés à la sous-section 4.6.7.

À la section suivante, nous présentons trois manières de regrouper les FL.

4.4 Regroupement des fonctions lexicales

Les FL peuvent être regroupées selon l'axe des relations lexicales (paradigmatiques vs syntagmatiques), selon la standardité et selon la compositionnalité.

4.4.1 Regroupement selon l'axe lexical

Tel que discuté à la section 1.8, les relations lexicales se présentent en deux axes : vertical (relations paradigmatiques) et horizontal (relations syntagmatiques).

De la même façon, les FL sont divisées en deux grands groupes : les fonctions lexicales paradigmatiques et les fonctions lexicales syntagmatiques.

Les FL paradigmatiques modélisent les relations paradigmatiques entre lexies. Voici des exemples de telles fonctions (Mel'čuk *et al.*, 1995) :

- Synonyme : Syn (*voiture*) = {*automobile*} ;
- Antonyme : Anti (*envoyer*) = {*intercepter*} ; Anti (*égal*) = {*inégal*} ;
- Généralisation : Gener (*gaz*) = {*substance* (*gazeuse*)} ; Gener (*pistolet*) = {*arme à feu*} ;
- Collectif : Mult (*navire*) = {*flotte*} ; Mult (*chien*) = {*meute*} ; Mult (*barbare*) = {*horde*}.

Les FL syntagmatiques modélisent les relations syntagmatiques entre lexies. En voici des exemples (Mel'čuk *et al.*, 1995) :

— Intensificateur : Magn (*peur*) = {*bleue*} ; Magn (*sou rir*) = {*atrocement*} ;

— Laudatif : Bon (*conseil*) = {*précieux*} ; Bon (*choix*) = {*heureux*} ;

— Instrumental : Instr (*téléphone*) = {*par* }} ; Instr (*outil*) = {*avec* [ART]} ;

— Verbe support : Oper_1 (*suprématie*) = {*détenir* [ART]} ; Oper_1 (*méfait*) = {*perpétrer* [ART]}.

Les relations entre lexies dans une collocation sont des relations syntagmatiques. Comme un des objectifs de cette thèse est la représentation de collocations, nous nous intéressons plutôt aux FL syntagmatiques. Voici des exemples de FL (Mel'čuk *et al.*, 1995) qui modélisent des collocations :

— Intensificateur : Magn (*amour*) = {*ardent*} ; Magn (*peur*) = {*bleue*} ;

— Confirmateur : Ver (*argument*) = {*valable*} ; Ver (*peur*) = {*justifiée*} ;

— Laudatif : Bon (*conseil*) = {*précieux*} ;

— Verbes supports :

- Oper_1(*remarque*) = {*faire* [ART]} ; Oper_2 (*danger*) = {*courir* [ART]} ;
- Func_1 (*responsabilité*) = {*incombe* [*à* N]} ; Func_1 (*problème*) = {*réside* [*dans* N]} ;
- Labor_{12} (*liste*) = {*mettre* [N *sur* ART]} ; Labor_{12} (*analyse*) = {*soumettre* [N *à* ART]}.

— Verbes de réalisation :

- Real_1 (*film*) = {*jouer* [ART]} ; Real_2 (*examen*) = {*réussir* [(*à*) ART]} ;
- Fact_0 (*rêve*) = {*se réalise*} ; Fact_2 (*médecin*) = {*traite* [N]} ;
- Labreal_{12} (*piège*) = {*prendre* [N *dans* ART]} ; Labreal_{12} (*balle*) = {*atteindre* [N *avec* ART]}.

4.4.2 Regroupement selon la standardité

Les fonctions lexicales sont classifiées comme *standard* et *non standard*. Dans cette thèse, nous ne nous intéressons qu'aux FL standard.

Pour être considérée comme standard, une FL *f* doit suivre les deux critères suivants (Mel'čuk, 1998) :

C1) *f* est définie pour un grand nombre d'arguments, c'est-à-dire que plusieurs lexies peuvent être bases de *f*. Le sens de *f* est suffisamment abstrait et général ;

C2) il y a un nombre relativement grand d'expressions qui sont des valeurs de *f*.

Toutes les FL citées dans les exemples précédents sont des FL standard : $Oper_i$, *Magn*, $Labreal_{ij}$, $Fact_i$, *Bon*, etc. Par exemple, la FL $Oper_i$ est applicable à plusieurs arguments et retourne plusieurs valeurs :

- Oper_1(*récital*) = {*donner* ART }
- Oper_1(*veste*) = {*subir, essuyer, ramasser* ART }
- Oper_1(*chantage*) = {*recourir, livrer, pratiquer, faire, exercer* ART }
- Oper_2(*chantage*) = {*subir* ART }
- Oper_2($repas_{I:2}$) = {*assister à* ART }
- etc.

Par contre, le sens « additionné de ... » est une FL non standard en français, car elle ne peut être appliquée qu'à un petit nombre d'unités lexicales (*café*, *fraises*, *thé*...) pour donner les expressions : *café crème*, *fraises à la crème* (et non **café à la crème*, **fraises crème*) ; *café au lait*, *café arrosé*, etc. (Mel'čuk, 1992)

4.4.3 Regroupement selon la compositionnalité

Selon la compositionnalité, une FL peut être *simple* ou *complexe*.

Les FL citées jusqu'ici sont des exemples de FL simples. Elles peuvent être combinées pour former des FL complexes, par exemple :

- CausOper_1(*crime*) = {*pousser à* ART }
- CausOper_2(*bouteille*) = {*embouteiller*}
- CausFunc_2(*bouteille*) = {*mettre en* }
- FinOper_2(*ami*) = {*perdre* ART }
- AntiBon(*dormir*) = {*insu samment*, *à la dure*}

Les exemples précédents montrent des FL complexes formées par juxtaposition, soit le cas le plus commun de FL complexe. Dans ce type de FL, il faut combiner les ASyntP des FL simples. Considérons, par exemple, la FL $\textit{CausOper}_1$.

L'$\textit{Oper}_1$ de *crime* est *commettre*. Lorsque on ajoute la FL *Caus*, pour former $\textit{CausOper}_1$, nous avons que l'ASyntP de « Caus(*commettre un crime*) = {*pousser à* } » est l'Agent de « pousser à commettre un crime ». En d'autres mots, « Caus(*commettre un crime*) = {*pousser à* } » y ajoute un ASyntP, qui est l'Agent de « pousser à commettre un crime ».

Il existe trois autres types de composition de FL simples, appelés configurations de FL :

- addition de FL : Magn+Oper_1(*sueur*) = {*nager dans la* }
- disjonction de FL (ou) : $\text{S}_3\text{\_or\_S}_1$(*débat*) = {*participant*}
- conjonction de FL (et) : $\text{S}_{loc}\text{\_and\_S}_2$(*résider*) = {*résidence*}

Mel'čuk *et al.* (1995) définissent la configuration de FL somme suit :

> Nous appelons configuration de fonctions lexicales une suite de FL simples qui ne sont pas syntaxiquement liées entre elles, mais qui ont le même mot-clé, cette suite ayant une valeur globale cumulative qui exprime de façon indécomposable le sens de la suite entière. (Mel'čuk *et al.*, 1995, p. 149)

Dans la modélisation de notre ontologie, pour simplifier le modèle, les trois types de configuration de FL sont traités comme des FL complexes. Une propriété a été créée pour distinguer les différents types de FL complexes, conformément à ce qui est montré à la sous-section 8.2.1.

4.5 Fonctions lexicales spéciales

Dans cette section, nous présentons des FL qui échappent au patron d'une FL conventionnelle. Nous les avons divisées en deux groupes, les types de spécialisation et les modificateurs des actants syntaxiques.

4.5.1 Types de spécialisation

Des indices sont utilisés pour indiquer que la valeur de la FL doit être la valeur usuelle ou prototypique de l'application de la FL à la base. En voici des exemples :

— $\mathrm{Oper}_1^{\mathrm{usual}}(\textit{sport}) = \{\textit{pratiquer}\ \mathrm{ART}\ \}$;

— $\mathrm{AntiBonReal}_1^{\mathrm{usual}}(\textit{alcool}) = \{\textit{boire, picoler}\}$;

— $\mathrm{S}_1^{\mathrm{prototype}}(\textit{ronfler}) = \{\textit{moteur}\}$;

4.5.2 Modificateurs des actants syntaxiques

Il y a trois cas spéciaux où il y a un changement par rapport aux actants syntaxiques :

— Changement de X (premier actant) ou Y (deuxième actant) : lorsqu'il y a une sorte de substitution d'un des actants. Comparez :

- $\mathrm{Oper}_1(\mathit{crime}_{I:a})$ = {*commettre* ART } (1)
- $\mathrm{Oper}_1{}^{\mathrm{XYChange}}(\mathit{crime}_{I:a})$ = {*tremper dans* ART } (2)

En (1), « *X commet un crime* ». En (2), il y a un autre premier actant (*X'*) qui agit comme une sorte de substitut pour le premier actant de référence : *X commet un crime* et *X' trempe dans le crime* pour substituer ou se joindre à *X*.

— Ensemble des individus : lorsque la FL est appliquée à un ensemble d'individus, par exemple :

- $\mathrm{Oper}_1{}^{\mathrm{setOfInd}}(\mathit{famille})$ = {*constituer, former* ART };

— L'actant *X* de la lexie a lui-même un actant. Par exemple, considérons la collocation *écouter la plage* (d'un disque), qui est modélisée par la FL Real_{11} (Real_1 de *1*), comme montré ci-dessous :

- $\mathrm{Real}_{1[1]}(\mathit{plage}_V)$ = {*écouter* ART };

Dans cet exemple, le premier actant de *plage* est *disque*, qui a lui-même un premier actant, qui est l'utilisateur du *disque*.

4.6 Indices des fonctions lexicales

Dans cette section nous présentons quelques indices qui sont utilisés dans certaines FL.

4.6.1 Indices actantiels

Nous expliquons dans cette sous-section le sens des indices numériques (*1,2,3,...*) sous certaines FL.

Considérons la fonction $Oper_i$. Pour cette fonction en particulier, les mots-clés (bases) sont toujours le complément d'objet direct (CO_{dir}) de la collocation (Mel'čuk *et al.*, 1995) :

— Oper_1 (*remarque*) = {*faire*} : « Jean a fait une remarque » *!* *remarque* est le CO_{dir} de la phrase ;

— Oper_2 (*danger*) = {*courir*} : « Jean court un danger » *!* *danger* est le CO_{dir} ;

— Oper_3 (*conseil*) = {*recevoir*} : « Jean recevra un conseil » *!* *conseil* est le CO_{dir}.

Pour la fonction $Oper_i$, l'indice indique toujours la fonction actantielle du sujet grammatical (SG) de la collocation (Mel'čuk *et al.*, 1995) :

— Pour Oper_1, l'indice *1* indique que le SG, *Jean*, est l'ASyntP_I (celui qui exécute l'action) de la base, *remarque* ;

— Pour Oper_2, l'indice *2* indique que le SG, *Jean*, est l'ASyntP_{II} (celui qui subit l'action) de la base, *danger* ;

— Pour Oper_3, l'indice *3* indique que le SG de la phrase, *Jean*, est l'ASyntP_{III} (celui qui est le destinataire de l'action) de la base, *conseil*.

Le SG, CO_{dir} et les autres CO sont les actants syntaxiques de surface. Donnons un dernier exemple pour mieux fixer ces concepts. Dans les phrases :

Jean donne un conseil à Marie. (1)
Marie reçoit un conseil de Jean. (2)

Nous avons :

— Phrase (1)

- SG = *Jean* ; CO_{dir} = *conseil* ; CO_{indir} = *Marie*
- ASyntP_{I} = *Jean* ; ASyntP_{II} = *conseil* ; ASyntP_{III} = *Marie*

— Phrase (2)

- SG = *Marie* ; CO_{dir} = *conseil* ; CO_{indir} = *Jean*
- ASyntP_{I} = *Jean* ; ASyntP_{II} = *conseil* ; ASyntP_{III} = *Marie*

Notons que les *SG* et les CO_{indir} des deux phrases sont l'inverse l'un de l'autre. Cependant, leurs *ASyntP* sont exactement les mêmes, car les deux phrases expriment le même sens : « X donne un Y à Z », même si dans la phrase (2) l'ordre des mots est différent de celui de la phrase (1).

Il faut noter qu'il s'agit de l'ASyntP du mot-clé (*conseil*) selon sa diathèse de base (organisation des rôles sémantiques de la voix active) (Polguère, 2011), même si elle n'est pas toujours reflétée dans la phrase.

4.6.2 Indication de degré de réalisation

Parfois, il n'est pas suffisant de seulement indiquer les ASem ou les ASyntP, car le fait peut se dérouler en différentes phases, en ayant les mêmes actants dans chaque phase. Par exemple, considérons les trois phrases suivantes :

— *Jean monte sur le cheval* ;

— *Jean chevauche le cheval* ;

— *Jean descend du cheval*.

Dans les trois actions, l'$ASem_1$ et l'$ASem_2$ sont toujours les mêmes, *Jean* et *cheval*, respectivement. Cependant, chaque action se déroule dans une phase différente. Pour indiquer différentes étapes des actions que possèdent les mêmes actants sémantiques potentiels, la TST utilise des chiffres romains (Jousse, 2010, p. 86).

Dans notre exemple, l'action de « monter sur le cheval » est représentée par la phase *I*, l'action de « chevaucher le cheval » est représentée par la phase *II* et l'action de « descendre du cheval » est représentée par la phase *III*.

Cette notion d'indication de phase est importante dans certaines FL, en particulier les FL de réalisation, comme $Real_i$ et $Fact_i$.

Les trois collocations dans les phrases précédentes (*monter sur le cheval*, *chevaucher le cheval* et *descendre du cheval*) sont modélisées par la FL $Real_1$ comme suit. À noter ici l'utilisation des chiffres romains pour indiquer les trois phases :

— $\mathrm{Real}_1^{\mathrm{I}}(\textit{cheval}) = \{\textit{monter sur}\ \mathrm{ART}\ \}$;

— $\mathrm{Real}_1^{\mathrm{II}}(\textit{cheval}) = \{\textit{chevaucher}\ \mathrm{ART}\ \}$;

— $\mathrm{Real}_1^{\mathrm{III}}(\textit{cheval}) = \{\textit{descendre de}\ \mathrm{ART}\ \}$;

4.6.3 Spécification spatiale

Les spécifications spatiales sont des indices utilisés avec la FL *Loc* pour indiquer la position : *in* (dans), *ab* (se déplaçant à partir de) et *ad* (se déplaçant pour se trouver dans) (Mel'čuk *et al.*, 1995). Exemples :

— $\mathrm{Loc}_{in}(\textit{armoire}) = \{\textit{dans}\ \mathrm{ART}\ \}$;

— $\mathrm{Loc}_{ab}(\textit{siège}) = \{\textit{par}\ \mathrm{le}\ \}$;

— $\mathrm{Loc}_{ad}(\textit{douche}) = \{\textit{à}\ \mathrm{la}\ \}$;

4.6.4 Spécification de circonstance

Les spécifications de circonstance sont des indices utilisés avec la FL *S* (nominalisation) pour indiquer (Mel'čuk *et al.*, 1995) :

— nom d'instrument (*instr*) : $\mathrm{S}_{instr}(\textit{crime}) = \{\textit{arme}\}$;

— nom de lieu (*loc*) : $S_{loc}(vote) = \{urne\}$;

— nom de moyen (*med*) : $S_{med}(chanter) = \{voix\}$;

— nom de manière (*mod*) : $S_{mod}(voter) = \{scrutin\}$;

— nom de résultat (*res*) : $S_{res}(endormir) = \{sommeil\}$;

4.6.5 Dimension de l'intensification

Ce sont des indices utilisés avec la FL *Magn* pour indiquer la dimension de l'intensification (Mel'čuk *et al.*, 1995). Il y a neuf de ces indices. En voici des exemples :

— comportement : $Magn_{behaviour}(ennemi) = \{mortel\}$;

— effet : $Magn_{effect}(marche) = \{triomphal\}$;

— épaisseur : $Magn_{thickness}(vêtement) = \{chaud, hiver, épais\}$;

— hauteur : $Magn_{height}(lac) = \{profond\}$;

— largeur : $Magn_{width}(vêtement) = \{ample, large\}$;

— longueur : $Magn_{length}(vêtement) = \{long\}$;

— puissance : $Magn_{strength}(tornade) = \{violent, puissant, intense, important\}$;

— taille : $Magn_{size}(vêtement) = \{grand\}$;

— vitesse : $Magn_{speed}(fleuve) = \{rapide\}$.

4.6.6 Dimension temporelle, de quantité et de genre

Ce sont des indices, représentés comme des exposants, utilisés avec la FL *Magn* pour indiquer l'intensification dans la quantité et dans le temps, avec la FL *Loc* pour indiquer la localisation dans le temps et avec la FL *Syn* pour indiquer le genre de la synonymie (Mel'čuk *et al.*, 1995). En voici des exemples :

— $Magn^{time}(ennemi) = \{irréconciliable\}$;

— $\mathrm{Magn}^{\mathrm{amount}}$(*manger*) = {*énormément, comme un ogre*} ;

— $\mathrm{Loc}_{in}{}^{\mathrm{time}}$(*visite*) = {*au cours de* ART ; *durant, pendant* ART } ;

— $\mathrm{Syn}_{more}{}^{\mathrm{sex}}$(*ami*) = {*amie*}.

4.6.7 Dégrée d'équivalence

Ce sont des indices utilisés, en particulier avec la FL *Syn* (synonymie), pour indiquer le degré d'équivalence, c'est-à-dire pour indiquer si les deux unités lexicales en question (la base et la valeur de la FL) ont une relation de synonymie d'intersection (*inter*), une relation dont le sens de la base est inclus dans celui de la valeur (*less*) ou une relation dont le sens de la base inclut celui de la valeur (*more*) (Mel'čuk *et al.*, 1995). Par exemple :

— Syn(*basé*) = {*situé*} ;

— Syn_{inter}(*anarchie*) = {*désordre*} ;

— Syn_{less}(*costume*) = {*vêtement*} ;

— Syn_{more}(*enfant*) = {*fils, fille*}.

Notons que les sens des FL Syn_{less} et Syn_{more} comprennent les sens de l'hyperonymie et de l'hyponymie, respectivement.

4.7 Classification sémantique des fonctions lexicales

Jousse (2010) présente quatre différentes classifications des fonctions lexicales : les classifications sémantique, syntaxique, combinatoire et pragmatique. Nous présentons ici la classification sémantique, qui est celle implémentée dans notre modèle.

Dans la classification sémantique, Jousse (2010) groupe les FL dans dix classes différentes : *action-événement*, *cause*, *élement-ensemble*, *équivalence*, *manière*, *localisation*, *opposition*, *participants*, *phase-aspect* et *qualificatif*. En plus de ces dix

classes, dans notre modèle, nous ajoutons deux autres classes : *verbe-sémantiquement-vide* et *verbe-support*.

Chaque classe est subdivisée en une ou plusieurs sous-classes. Par exemple, la classe *action-événement* est divisée en huit sous-classes, comme suit :

tentative, *création*, *diminution-dégradation*, *manifestation*, *utilisation-typique*, *augmentation-amélioration*, *disparition-fin d'existence* et *non-fonctionnement*. En plus de ces huit sous-classes, nous ajoutons la classe *imminence*.

Un autre exemple : la classe *localisation* est divisée en deux sous-classes : *lieu-typique* et *spatial-temporel*.

Chaque FL a au moins un sens prédicatif qui lui est associé, donc chaque FL est classifiée dans une ou plusieurs perspectives sémantiques. Par exemple, la FL *Magn* (intensification) est associée à la classe *qualification* et à la sous-classe *intensité*. La FL *Bon* est aussi associée à la classe *qualification*. Par contre, elle est pour sa part associée à la sous-classe *jugement*.

4.8 Règles de paraphrasage de la théorie Sens-Texte

Mel'čuk (1992) et Milićević (2007) présentent les *règles de paraphrasage* de la TST. Nous montrons ici certaines de ces règles et comment les FL sont utilisées pour la modélisation de paraphrases.

Mel'čuk (1992) divise ces règles en deux grands groupes : les règles lexicales de paraphrasage (54 règles) et les règles syntaxiques de paraphrasage (29 règles). Nous ne donnons ici que des exemples de règles lexicales.

Les règles lexicales de paraphrasage sont sous-divisées en deux groupes : équivalence sémantique et implication sémantique.

Les exemples suivants sont tirés de (Mel'čuk, 1992). C_0 dénote le mot-clé de départ, sur lequel la règle est appliquée et qui est le mot-clé de la FL correspondante (Mel'čuk, 1992).

La règle de paraphrasage la plus simple est la *substitution synonymique*. Les règles R1, R2, R3a et R3b sont des exemples de règles lexicales d'équivalence sémantique et la règle R4 est un exemple de règle lexicale d'implication sémantique.

R1 - Substitution synonymique : C_0 , $Syn(C_0)$. Exemple :

A. « Ce phénomène s'explique facilement »

m

B. « Ce phénomène s'explique sans difficulté »

Où : C_0 = *facilement* ; et Syn(*facilement*) = {*sans di culté*}

R2 - Substitution conversive à deux arguments : C_0 , $Conv_{21}(C_0)$. Exemple :

A. « Je crains les conséquences »

m

B. « Les conséquences m'effraient »

Où : C_0 = *craindre* ; et $Conv_{21}$(*craindre*) = *e rayer*

R3a - Fissions à verbe support : C_0 , $S_0(C_0)$ $Oper_1(S_0(C_0))$. Exemple :

A. « Jean nous a bien accueillis »

m

B. « Jean nous a réservé un bon accueil »

Où : C_0 = *accueillir* ; S_0(*accueillir*) = {*accueil*} ; et $Oper_1$(*accueil*) = {*réserver* DET }

R3b - Fissions à verbe support : C_0 , $S_0(C_0)$ $Oper_2(S_0(C_0))$. Exemple :

A. « Jean nous a bien accueillis »

m

B. « Nous avons trouvé chez Jean un bon accueil »

Où : C_0 = *accueillir* ; S_0(*accueillir*) = {*accueil*} ; et $Oper_2$(*accueil*) = {*trouver* DET chez N}

R4 - Implication sémantique : $CausX(C_0)$) $IncepX(C_0)$. Exemple :

A. « Jean a mis en marche le moteur »

m

B. « Le moteur a démarré »

Où : C_0 = *moteur* ; X = $Fact_0$; $CausFact_0$(*moteur*) = {*mettre en marche*} ; et $IncepFact_0$(*moteur*) = {*démarrer*}

4.9 Propriétés des fonctions lexicales

Kolesnikova (2011, p. 68–69) présente certaines propriétés importantes des FL standards pour les applications informatiques :

— Les FL standards sont universelles. Elles représentent des relations sémantiques qui sont présentes dans toutes les langues. Cela nous permet de les utiliser pour la construction des représentations en plusieurs langues pour un éventuel alignement multilingue. Elles peuvent être utilisées pour les applications de traduction automatique, la recherche d'information multilingue, l'alignement des ontologies en différentes langues, etc. ;

— Les FL sont idiomatiques. Cela permet la représentation des sens « non typiques » qui émergent seulement lorsque certains mots sont trouvés ensemble. Par exemple, en anglais, on peut dire *to know firmly* (lit. savoir fermement). Dans cette expression, le sens du mot *firmly* est idiomatique. On peut utiliser la FL Magn (intensificateur) pour le représenter : Magn (*know*) = *firmly* ;

— Les FL sont paraphrasables. Le fait d'être paraphrasables est utile pour la traduction (traduire un mot au lieu d'une expression, par exemple) et pour la recherche d'information. Par exemple, *faire une analyse* = *analyser*. Oper_1 (*analyse*) = {*faire*} ; V (*analyse*) = *analyser* ;

— Les FL sont sémantiquement diverses. Parfois, les valeurs des mêmes FL avec les mêmes arguments ne sont pas les mêmes. Par exemple, en anglais, Magn (*know*) = {*firmly*}. Cependant, il y a aussi les possibilités : Magn (*know*) = {*deeply*/*profoundly*} ou Magn (*know*) = {*broadly*/*extensively*}. Or, les lexies *firmly*, *deeply* et *broadly* ne sont pas des synonymes dans n'importe quel contexte. Pour représenter ces différents sens, on peut, par exemple, nommer les fonctions comme Magn_1 (*know*) = {*firmly*} ; Magn_2 (*know*) = {*deeply*/*profoundly*} ; Magn_3 (*know*) = {*broadly*/*extensively*}. Et les sens attribués à $Magn_1$, $Magn_2$ et $Magn_3$ sont ensuite les mêmes en différentes langues.

4.10 Avantages des fonctions lexicales

Un des avantages des FL est son pouvoir de systématisation de la connaissance sur les collocations (et sur d'autres relations syntaxiques/sémantiques entre des lexies en général). Présentement, plus de 70 FL ont été identifiées dans plusieurs langues humaines (Kolesnikova, 2011).

Un autre avantage est la possibilité de représenter les collocations d'une manière indépendante de la langue et la possibilité d'utiliser cette caractéristique dans certaines tâches du traitement automatique des langues, comme la traduction automatique et la recherche d'information translingue.

Par exemple, si en anglais, on prend la FL *Oper*$_1$ avec l'argument « attention » nos avons : Oper$_1$ (*attention*) = {*pay*} (*pay attention*). En français, la collocation équivalente à *pay attention* est *faire attention*. Cependant, la traduction littérale vers le français serait : **payer attention*. C'est un des problèmes que les collocations posent pour la traduction automatique.

Cependant, on peut représenter en français cette collocation comme la lexie *faire* étant le résultat de l'application de la fonction *Oper*$_1$ sur l'argument *attention*, et représenter la même collocation en anglais comme le mot *pay* étant le résultat de l'application de la même fonction sur l'argument *attention*. Et, de la même manière, on peut représenter la collocation équivalente en portugais, *prestar atenção* (lit.*rendre attention*).

La figure 4.1 montre un exemple d'alignement de la représentation de la collocation *faire attention* en trois langues : français, anglais et portugais.

Ensuite, si on veut traduire la collocation *pay attention* vers le français, on pourrait traduire la lexie *attention* par la lexie française *attention*. Ce type de traduction ne pose pas de problème, car on traduit seulement la base de la collocation, qui

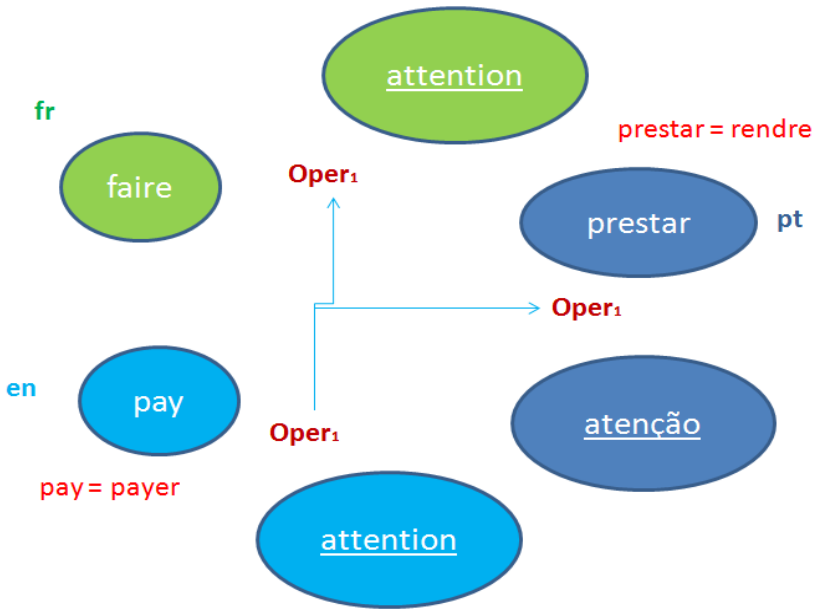

Figure 4.1: Alignement de la représentation de la collocation *faire attention* en trois langues

généralement garde son sens littéral. Dans une collocation, c'est le collocatif (dans notre exemple, le verbe), qui pose problème pour la traduction, car son sens dans la collocation est souvent idiomatique.

Après la traduction de la lexie *attention* en anglais vers *attention* en français, on cherche dans la représentation en français quel est le résultat de l'application de la même fonction ($Oper_1$), utilisée pour représenter la collocation en anglais, sur la lexie *attention*, et cela nous donne comme résultat l'ensemble {*faire*, *porter*, *prêter*}.

4.11 Conclusion

Nous avons vu dans ce chapitre la définition, les caractéristiques, des applications et des avantages de l'utilisation des fonctions lexicales.

Les fonctions lexicales sont un outil linguistique puissant pour la représentation

des relations lexicales et à notre connaissance, le seul à modéliser systématiquement les relations syntagmatiques.

Au chapitre suivant, nous présentons le concept de réseau lexical et nous montrons quelques exemples de réseaux lexicaux, en particulier le Réseau lexical du français (RLF), qui est basé sur les fonctions lexicales.

CHAPITRE V

RÉSEAUX LEXICAUX

5.1 Introduction

Un réseau lexical (RL) est une représentation par graphe du lexique (ensemble des mots d'une langue) et des connexions entre les mots (relations lexicales) qui forment le lexique.

On trouve ici une question de terminologie. Faut-il parler de relations lexicales ou de relations sémantiques ? Le problème est qu'il est difficile de dissocier les deux, dès lors que les unités lexicales portent aussi des sens. Eluerd (2000) a bien explicité la question :

> La sémantique, voisine encombrante de la lexicologie ? Il y a là un paradoxe indéfendable. Le partage traditionnel des tâches de la lexicologie en morphologie lexicale et sémantique lexicale laisse assez entendre qu'on ne peut imaginer une lexicologie coupée de la sémantique, indifférente à l'analyse du sens (Eluerd, 2000, p. 30).

Selon Grossmann (2011), la notion de RL peut être liée à la notion de lexique mental. Il ajoute :

> Plutôt que comme un « stock » ou un « trésor » semblable à un dictionnaire, on préfère aujourd'hui se représenter ce lexique comme le

> résultat, toujours évolutif, de processus créant des liens sémantiques nouveaux.
> Le lexique mental est organisé à partir du sens plutôt que de la forme des mots, mais la parenté morphologique lorsqu'elle est associée au sens, joue un rôle important pour faciliter les associations et la mémorisation (Grossmann, 2011, p. 174).

Les RL suivent le modèle relationnel de la représentation lexicale et sont basés sur les travaux psycholinguistiques de Quillian (1968) portant sur la représentation du lexique mental.

Il y a d'autres modèles de représentation lexicale. Jousse (2010), par exemple, en identifie quatre types — componentiel, génératif, décompositionnel et relationnel —, en faisant une comparaison entre ces différents modèles. Nous présentons ici le modèle relationnel, qui est celui lié à cette thèse.

Le modèle relationnel suit l'approche appelée *meaning postulate* (Fodor *et al.*, 1975; Murphy, 2003), selon laquelle le sens d'un mot n'émerge pas de sa décomposition en sous-parties lexicales, mais plutôt à partir de ses relations avec d'autres mots (Murphy, 2003). Par exemple, au lieu d'inclure *félin* comme une partie du sens de *chat*, on établit que, si un animal est un chat, alors il est un félin (relation de subsomption, hiérarchique ou relation hyperonymie/hyponymie).

C'est à cause de la relation de subsomption qu'on considère les réseaux lexicaux qui suivent le modèle relationnel comme étant des ontologies. Par exemple, Hirst (2004) affirme ce qui suit par rapport à la relation entre un lexique (computationnel), comme WordNet, et une ontologie :

> The obvious parallel between the hypernymy relation in a lexicon and the subsumption relation in an ontology suggests that lexicons are very similar to ontologies. It even suggests that perhaps a lexicon, together with the lexical relations defined on it, is an ontology (or is a kind of

ontology in the ontology of ontologies) (Hirst, 2004, p. 8).

Cependant, Hirst (2004) continue sa discussion sur ce sujet pour conclure qu'un réseau lexical, en dépit des similarités, n'est pas une ontologie. Son principal argument est qu'une ontologie est un ensemble de catégories liées par des relations, alors qu'un réseau lexical représente les mots et sens d'une langue et les relations entre mots et sens. Et chaque langue catégorise le monde d'une manière différente. Cela cause les différences identifiées par Hirst (2004) et présentées ici :

— L'intersection entre les catégories d'une ontologie est normalement vide, tandis que dans un lexique, il y a des presque synonymes (par ex. : *erreur*, *faute*, *méprise*, *faux pas*, etc.).
— Il y a des lacunes dans les lexiques : plusieurs concepts ou catégories ne sont pas lexicalisés.
— Il y a des distinctions de sens dans les lexiques qui sont arbitraires d'un point de vue ontologique. Ou encore, la catégorisation est différente dans chaque lexique. Par exemple, en français il y a une distinction entre *fleuve* et *rivière*, tandis qu'en anglais, cette distinction n'existe pas.

Parmi les RL les plus connus, on compte : WordNet (Fellbaum, 1998), FrameNet (Fillmore, 1977) et MindNet (Richardson *et al.*, 1998). Jousse (2010, p. 25–56) présente une comparaison entre différents RL.

Finalement, nous soulignons que des travaux sur l'utilisation de bases de données et de réseaux lexicaux pour l'apprentissage profond existent. Par exemple, Bordes *et al.* (2011) proposent une architecture pour transformer l'information contenue dans un graphe lexical vers les vecteurs de mots. Ils appliquent cette architecture à WordNet et à Freebase[1].

1. https ://developers.google.com/freebase/

Nous présentons dans les sections suivantes quelques exemples de RL : WordNet, FrameNet, MindNet, ConceptNet (Liu et Singh, 2004) et le Réseau Lexical du Français (RLF) (Lux-Pogodalla et Polguère, 2011).

5.2 WordNet

WordNet (Fellbaum, 1998) est sans doute le RL le plus connu et le plus utilisé par la communauté du TALN. Le projet WordNet a été motivé par des études sur l'organisation des connaissances dans le cerveau humain (Fellbaum, 1998) et depuis sa création, il a été utilisé dans le cadre de plusieurs travaux liés aux domaines du TALN, de la psycholinguistique, de l'intelligence artificielle, etc.

WordNet est organisé par *synsets*, qui peuvent être considérés comme des ensembles de mots « plus ou moins » synonymes. Cela veut dire que les mots dans un *synset* sont des synonymes dans certains contextes (Fellbaum, 1998). En gros, à chaque *synset* correspond un concept (Schwab *et al.*, 2007).

Différents chercheurs considèrent WordNet comme des relations entre concepts (relations sémantico-conceptuelles) ou comme des relations entre lexies (relations lexicales) (Gangemi *et al.*, 2003). Au début, WordNet était purement relationnel, ce qui a changé après l'insertion de gloses (c'est-à-dire de définitions textuelles) pour définir chaque *synset* (Kolesnikova, 2011).

Les *synsets* sont connectés par des relations, comme l'antonymie et l'hyperonymie. L'hyperonymie est une relation entre lexies où le champ lexical (un ensemble de lexies appartenant à la même catégorie syntaxique et liés par leur domaine de sens) d'un mot contient le champ lexical de l'autre lexie. L'hyponymie est la relation inverse de l'hyperonymie. Par exemple, *mammifère* est un hyperonyme de *félin*, qui est un hyperonyme de *chat*, et *chat* est un hyponyme de *félin*, qui est un hyponyme de *mammifère*.

Ces relations sont transitives : si *mammifère* est un hyperonyme de *félin*, qui est un hyperonyme de *chat*, alors *mammifère* est aussi un hyperonyme de *chat*. Finalement, deux *synsets* sont co-hyponymes s'ils ont un hyperonyme en commun et s'ils ne sont pas l'hyponyme l'un de l'autre, par exemple *chat* et *lion*, qui ont l'hyperonyme *félin* en commun.

Dans le contexte du Web sémantique, l'hyperonymie et l'hyponymie sont comparables à des relations *is-a* (*est-un*). Par exemple, *le chat est un* (*is-a*) *félin*. Ces relations sont la base pour la construction des taxonomies et des ontologies.

Une des critiques portées à WordNet est le fait qu'il n'implémente que des relations paradigmatiques (synonymie, antonymie, hyperonymie, etc.). Les relations syntagmatiques entre synsets y sont absentes (Fellbaum, 1998; Schwab *et al.*, 2007; Jousse, 2010).

Un autre problème de WorldNet est sa macrostructure : son organisation par *synsets* empêche la création des liens pour représenter les collocations, car ces dernières sont liées à des lexies individuelles.

Nous croyons que cette absence est un problème et que la recherche dans les domaines du TALN et de la psycholinguistique pourrait profiter de l'enrichissement de WordNet avec des relations syntagmatiques, qui sont le type de relation qui connecte les lexies dans une collocation.

Finalement, WordNet est basé sur des études portant sur la cognition. Cependant, il a été très peu utilisé pour ce type d'études :

> On the negative side, while WordNet aimed at being a cognitively motivated model, it is being used mainly to support studies and applications that have little to do with cognition. One probable cause of this state of affairs is that WordNet's ontological structure is not as cognitively relevant as it was expected to be by its designers. Another

> explanation is to be found in the fact that WordNet became rapidly so popular in the Natural Language Processing community that the focus of the project shifted at an early stage from psycholinguistics to computer applications. (Polguère, 2014, p. 397).

5.3 FrameNet

FrameNet (Baker *et al.*, 1998; Ruppenhofer *et al.*, 2006) est un projet dont l'objectif est de documenter la gamme des possibilités combinatoires sémantiques et syntaxiques (valences), de chaque mot en chacun de ses sens.

L'appariement d'un mot avec un sens est appelé *unité lexicale*. Pour les mots polysémiques, chaque sens correspond à un cadre sémantique différent.

FrameNet est basé sur l'idée de cadres sémantiques, qui sont des structures conceptuelles décrivant une situation, un objet ou un événement, avec ses participants et propriétés (Ruppenhofer *et al.*, 2006). *Cadres sémantiques* est aussi le nom d'une théorie pour la représentation des sens (Padó, 2007).

Liés à chaque sens (cadre), il y a les verbes qui expriment ce sens, connus comme *Frame Evoking Elements*, et leurs réalisations syntaxiques possibles. Chaque cadre montre aussi les éléments qui participent à chaque situation sémantique, appelés les *éléments du cadre*. Chaque élément porte un rôle sémantique.

Par exemple, le cadre *Apply_Heat* décrit une situation avec les éléments du cadre *Cook*, *Food* et *Heating_Instrument* et il est évoqué par des verbes comme *bake*, *boil*, *steam*, etc. Ces verbes sont des unités lexicales.

Chaque unité lexicale est présentée avec une définition et des exemples réels d'application, extraits des corpus textuels. À la différence de la lexicographie (construction de dictionnaires), où la notation des sens est faite mot à mot, dans FrameNet, le travail de notation est fait par cadres. Pour chaque cadre sont trouvées les unités

lexicales qui l'évoquent. Une unité lexicale peut être liée à plusieurs cadres.

L'implémentation du FrameNet a commencé en 1998 (Baker *et al.*, 1998) pour la langue anglaise, et aujourd'hui est en cours le développement de fragments en français, portugais, espagnol, japonais, allemand et hébreu, entre autres.

5.4 MindNet

MindNet (Richardson *et al.*, 1998) est un réseau sémantique construit automatiquement à partir de dictionnaires électroniques (le *Longman Dictionary of Contemporary English, LDOCE* et le *Webster's 7th New Collegiate Dictionary*), avec l'utilisation d'analyseurs syntaxiques.

Un des avantages de MindNet est la diversité de ses relations sémantiques, appelées « semrels ». On cite, par exemple : *attribute*, *cause*, *color*, *deep-object*, *deep-subject*, *equivalent*, *hypernym*, *location*, *purpose*, *size*, *time*, *user*, etc.

Les relations sont obtenues à partir de l'analyse syntaxique des définitions des dictionnaires. Par exemple, la figure 5.1 montre quelles sont les relations extraites à partir de la définition de la lexie *car* (voiture).

```
car       hyp >        vehicule
          part >       wheel
    <      Tobj         drive       means >      motor
           Purp >       carry       Tobj >       people
```

Figure 5.1: Représentation des relations sémantiques liées à la lexie *car* dans MindNet (Richardson *et al.*, 1998)

Après la création des *semrels*, les relations sont inversées et combinées avec d'autres relations, pour en représenter le sens inverse. Par exemple, la représentation de *driver* (conducteur) peut être combinée avec la représentation inverse de *car* (voiture).

Finalement, d'autres relations ont été créées par déduction logique, en utilisant des relations de synonymie et d'hyperonymie. Par exemple, des relations entre *car* et *truck* (camion) ont été créées en sachant que ces mots ont un hyperonyme en commun (*vehicule*).

5.5 ConceptNet

Il est important de noter qu'il y a des types de réseaux, autres que les RL, qui représentent des concepts et des relations sémantiques entre concepts. Parmi ces réseaux, nous mentionnons ConceptNet [2] (Liu et Singh, 2004).

ConceptNet est un réseau sémantique dont les nœuds sont des concepts dits de *sens commun* (*commonsense knowledge*). Sa structure est similaire à celle de WordNet, mais il est plus riche en connexions. ConceptNet en est déjà à sa cinquième version (Speer et Havasi, 2013).

À titre d'exemples de concepts de sens commun, les auteurs citent : *wake up in the morning* (*se lever le matin*), *eat breakfast* (*prendre le petit déjeuner*), *full stomach* (*ventre plein*), *wake up* (*se réveiller*), *in house* (*dans la maison*), *kitchen table* (*table de cuisine*), etc.

Pour la construction des concepts, une plateforme en ligne a été créée et des phrases du type *The e ect of eating food is...* (*l'e et de manger de la nourriture est...*) ont été présentées à des collaborateurs inscrits. Les réponses ont été collectées et 700 000 concepts ont été obtenus, réduits à 300 000 après l'application des filtres et des règles d'extraction de patrons.

Dans ConceptNet, les types de relations sont aussi très variés. Par exemple, il y a les relations :

---

2. http ://conceptnet.io/

— *prerequisiteOf* (*condition préalable*) : « wake up in the morning » *prerequisiteOf* « eat breakfast » ;

— *generalisation* (*généralisation*) : « wake up in the morning » *generalisation* « *wake up* » ;

— *usedFor* (*utilisé pour*) : « kitchen table » *usedFor* « eat breakfast » ;

— *locationOf* (*localisation de*) : « in house » *locationOf* « kitchen table ».

L'avantage principal de ConceptNet est sa richesse en termes de concepts et de relations conceptuelles. Cependant, les concepts sont très spécifiques et leur quantité est trop imposante pour pouvoir en faire une généralisation et une classification.

De plus, la façon de construire les concepts est censée être distribuée à des milliers de personnes, mais les réponses sont induites dès le début. Par exemple, quand la question initiale est « The effect of eating food is... », on s'attend déjà à ce que la réponse soit « to fill the stomach » (remplir le ventre), d'où le concept *full stomach* obtenu.

Finalement, les concepts créés ne sont pas des concepts lexicalisés.

5.6 BabelNet

BabelNet [3] est un réseau lexical multilingue construit automatiquement à partir de WordNet et de Wikipédia (Navigli et Ponzetto, 2012). Des techniques d'apprentissage machine — basées sur *bag-of-words* (sac des mots) [4] et sur des analyses de graphes — son utilisées pour son enrichissement.

Presque la totalité des mots dans BabelNet (environ 99%) sont des noms, pour seulement 1% de verbes, d'adjectifs et d'adverbes. Tout comme WordNet, Babel-

3. http ://babelnet.org/

4. Représentation des documents par la fréquence des mots qu'ils contiennent

Net contient uniquement des relations paradigmatiques.

Dans sa forme initiale, le projet BabelNet a été construit pour six langues : l'anglais, le catalan, le français, l'allemand, l'italien et l'espagnol (Navigli et Ponzetto, 2012).

5.7 Réseau Lexical du Français (RLF)

Dans cette section, nous présentons le Réseau lexical du français (RLF) (Lux-Pogodalla et Polguère, 2011), qui est, à notre connaissance, le seul RL basé sur la TST. Le RLF a été construit manuellement par une équipe lexicographique d'environ 15 personnes, dans le cadre du projet RELIEF [5].

Lux-Pogodalla et Polguère (2011) expliquent que les stratégies lexicographiques utilisées pour faire l'extraction de l'information linguistique à partir des corpus linguistiques sont basées sur la lexicographie explicative et combinatoire (Mel'čuk *et al.*, 1995). Ils ont aussi largement utilisé le *Trésor de la Langue Française informatisé* [6] (Dendien et Pierrel, 2003) comme base de données lexicale à partir de laquelle des informations lexicographiques sont extraites. Ce réseau prend aussi comme base le dictionnaire DiCo (Polguère, 2003).

Leur projet est la création d'un RL, comme WordNet et FrameNet (Baker *et al.*, 2003). Cependant, ils proposent une représentation globale de toutes les propriétés lexicales contenues par les dictionnaires : définitions lexicographiques, caractéristiques grammaticales, combinaisons lexicales (par ex. fonctions lexicales), etc.

Dans ce réseau, les unités lexicales sont connectées par des relations sémantiques et combinatoires. Les relations sémantiques sont représentées par des FL para-

---

5. http ://www.atilf.fr/spip.php ?article908

6. http ://www.atilf.fr/spip.php ?rubrique77

digmatiques qui connectent deux unités lexicales. Voici des exemples de ce type de relation : synonymie, antonymie, conversion verbale (*célébration ! célébrer*), conversion adverbiale (*lent ! lentement*), etc. Les relations combinatoires sont représentées par les FL syntagmatiques.

Pour chaque entrée du réseau, il y a :

— des caractéristiques grammaticales : partie du discours, inflexions, etc. ;

— une définition (et pour chaque unité lexicale utilisée dans la définition, il y a un lien qui la connecte à sa propre définition, soit une définition déjà existante dans le réseau, soit une définition en attente) ;

— les liens de fonctions lexicales contrôlés par l'unité lexicale ;

— des exemples d'utilisation ;

— une section de phraséologie, qui inclut les phrasèmes qui contiennent l'unité lexicale.

Pour représenter les collocations, l'approche proposée est l'insertion des FL dans chaque entrée pour créer des liens entre les unités lexicales qui forment des collocations, de la même manière implémentée dans le dictionnaire DiCo (Polguère, 2003).

Nous illustrons avec un exemple comment le RLF stocke des informations sur des FL, des unités lexicales et l'information sur les relations syntagmatiques (et paradigmatiques) entre des unités lexicales. À la section 8.6, nous montrons comment le même exemple est représenté en utilisant notre ontologie métalinguistique.

Nous montrons dans les paragraphes suivants comment la collocation *porter un vêtement* (Real_1(*vêtement*) = {*porter DET* }) est représentée dans le RLF.

Le tableau 5.1 montre un extrait d'un tableau où sont stockées des informations

Tableau 5.1: Représentation des fonctions lexicales dans le RLF

id_lf	name_lf	family_id
2	Syn	3
168	Oper_1	60
228	Real_1	69

Tableau 5.2: Données représentant les mots *porter* et *vêtement* et quelques-uns de leurs sens dans le RLF

(a) Tableau des vocables

vocable_id	name
28885	porter
29820	vêtement

(b) Tableau des sens

sense_id	vocable_id	sense_num
28883	28885	I.1
38555	28885	IV
29818	29820	I.2
38335	29820	I.1
38336	29820	II
38342	29820	III.1
48705	29820	III.2

sur les FL. Il y a trois FL représentées : Syn, $Oper_1$ et $Real_1$. Le tableau 5.2(a) montre la représentation des unités lexicales *porter* et *vêtement* et le tableau 5.2(b) la représentation de certains de leurs sens, ceux qui participent à des collocations.

Le tableau 5.3(a) montre la combinaison de la FL $Real_1$ ($lf\_id = 228$) avec l'unité lexicale *vêtement_I.2* ($sense\_id = 29818$). Le tableau 5.3(b) montre la relation entre la paire ($Real_1$, *vêtement_I.2*) et *porter_IV* ($sense\_id = 38555$) pour représenter la collocation *porter un vêtement*.

Tableau 5.3: Données représentant la relation entre les unités lexicales *porter_IV* et *vêtement_I.2* pour former la collocation *porter un vêtement* dans le RLF

(a) Relation entre une FL et un sens agissant comme une base dans une collocation

sense_lf_id	sense_id	lf_id
11264	29818	228

(b) Relation entre le pair FL/base et la cible (valeur) d'une collocation

sense_lf_id	target_id
11264	38555

Nous observons par cet exemple que, dans le RLF, les informations lexicales et combinatoires ne sont pas facilement transférables. En outre, elles ne peuvent pas être appliquées directement à la représentation des FL et des collocations dans d'autres réseaux ou ressources lexicales, dans d'autres langues.

5.8 Conclusion

Nous avons vu dans ce chapitre la notion de RL relationnel et nous avons donné quelques exemples de réseaux, comme WordNet, FrameNet, MindNet et le Réseau lexical du français.

WordNet est le RL le plus connu et utilisé par la communauté scientifique liée au TALN. Sa construction est basée sur des études psycholinguistiques sur le lexique mental.

En dépit du succès de WordNet, la façon dont les mots y sont organisés en synsets et sa hiérarchie conceptuelle, ont reçu des critiques (Fontenelle, 2012, p. 438-439). De plus, son manque de connexions syntagmatiques est considéré comme un problème (Schwab *et al.*, 2007).

Le RLF, qui est basé sur les FL, règle ce problème de manque de connexions syntagmatiques. Cependant, la façon dont les données sont stockées rend difficile le partage des informations sur les FL et sur les collocations d'une façon indé-

pendante de la structure du réseau. De plus, cette structure ne permet pas une intégration avec des ressources sur le Web.

Pour pallier ce problème, nous avons développé une ontologie lexicale pour représenter les FL et l'avons ensuite utilisée pour représenter le RLF sous la forme d'une ontologie. Dans ce format, le RLF sera plus facilement connecté à d'autres ressources sur le Web, comme des dictionnaires, d'autres réseaux lexicaux, etc.

Avant de présenter notre formalisme pour la représentation des FL et des relations lexicales sur le Web, nous parlons, au chapitre suivant, de ceux dont nous avons besoin pour accomplir cette tâche : les formalismes du Web sémantique.

CHAPITRE VI

WEB SÉMANTIQUE ET ONTOLOGIES

6.1 Introduction

La croissance d'Internet, de la quantité d'informations et la nécessité d'accéder à celles-ci afin de répondre à des questions complexes ont augmenté l'importance d'avoir des représentations des connaissances axées sur le sens. Cela a donné lieu à la création du Web sémantique.

Hendler et van Harmelen (2008) donnent la définition suivante du Web sémantique :

> The Semantic Web is an extension of the current World Wide Web in which information is tied to machine-readable metadata, making it easy to exchange, integrate and process data in a systematic, machine-automated manner (Hendler et van Harmelen, 2008, p. 823).

Dans le Web d'aujourd'hui, la majorité de l'information est présentée d'une façon compréhensible pour des êtres humains, mais peu compréhensible pour une machine. Par exemple, dans une base de données, les informations sont enregistrées de manière à ce qu'un système puisse les extraire, faire des inférences, des calculs, etc. Par contre, dans un fichier *HTML*, l'information n'est pas encodée de façon à ce que le sens soit accessible à une machine. Considérons le fragment suivant, encodé en *HTML* :

```
<html>
    <body>
        <text>
            Le Canada, dont la capitale
            est Ottawa, est un pays de
            36.000.000 d'habitants.
        </text>
    </body>
</html>
```

Et maintenant cette représentation de la même information :

```
<pays>
    <nom-pays>Canada</nom-pays>
    <capitale-pays>Ottawa</capitale-pays>
    <population>36.000.000</population>
</pays>
```

Le deuxième encodage est beaucoup plus facile à traiter par une machine. Ce type de langage est un langage de métadonnées, car il donne des informations sur l'information représentée. La métadonnée essaye de capturer une partie du sens des données, d'où l'adjectif « sémantique » dans l'expression « Web sémantique » (Antoniou et van Harmelen, 2004).

L'objectif du Web sémantique est donc de représenter les connaissances de façon à ce qu'elles soient accessibles et compréhensibles pour les êtres humains et pour les machines, en changeant l'architecture actuelle du Web afin d'inférer plus facilement les relations entre les différents objets.

Dans ce contexte, il est aussi important de représenter les connaissances spécifiques à chaque domaine. Chaque domaine contient son vocabulaire propre, où un mot représente un concept, une idée ou une entité. Les mots qui sont spécifiques à un domaine sont appelés *termes*, et l'ensemble des termes d'un domaine forme sa terminologie.

Souvent, de telles terminologies sont représentées par des graphes où les nœuds sont des concepts et les arcs sont des relations entre les concepts. Ces représentations sont construites de manière à ce que la recherche d'information soit plus efficace et que les relations entre les termes soient correctement encodées. Ces bases de données terminologiques doivent fonctionner comme des répertoires que l'on peut utiliser pour répondre à des questions, faire de la recherche d'information, aider dans la construction de dictionnaires, etc.

Ces représentations des connaissances sont appelées *ontologies informatiques*. Puisque certains langages créés pour la construction d'ontologies sont basés sur la logique du premier ordre, nous la présentons dans la section suivante, avant d'introduire les ontologies et les langages utilisés dans leur construction.

6.2 Logique du premier ordre

La logique du premier ordre (Lifschitz *et al.*, 2008), ou calcul des prédicats, a été proposée par Frege (1948). Elle est une tentative de modéliser mathématiquement le raisonnement humain.

Des variables sont utilisées pour représenter des éléments (entités génériques) sur lesquels on fait des affirmations afin d'arriver à des conclusions. Les prédicats représentent des relations entre éléments et il y a des connecteurs logiques (*et*, *ou*, *négation*) entre ceux-ci. Finalement, deux quantificateurs peuvent être appliqués aux variables : le quantificateur universel « quel que soit » et le quantificateur existentiel « il existe au moins un... tel que ».

Exemple d'affirmation en logique du premier ordre : $(8x)\ homme(x)\ (9y)\ homme(y)\ pere(y;x)$ *! quel que soit l'homme, il existe au moins un autre homme tel qu'il soit son père.*

Où : x et y sont des variables ; $homme()$ est un prédicat unaire et $pere()$ est

un prédicat binaire ; $\forall$ est le quantificateur universel et $\exists$ est le quantificateur existentiel.

Différents langages, basés sur la logique du premier ordre ont été créés, nous en présenterons un, la logique de description.

6.2.1 Logique de description

La logique de description (*description logic* (DL)) (Baader *et al.*, 2009) est un langage basé sur la logique du premier ordre créé pour la représentation des connaissances d'une manière structurée et formelle.

Il y a trois notions importantes dans la logique de description :

- concept : un concept est une classe d'individus. Nous pouvons dire que ce que la DL nomme « concept » est la notion de « catégorie » ;
- rôle : un rôle est une relation binaire entre concepts ;
- individu : un individu est un élément sur lequel des rôles peuvent être appliqués.

Par exemple, en DL on peut définir le concept « un homme célibataire qui est pilote et dont tous les frères sont des pilotes ou des ingénieurs » comme suit :

$$Humain \sqcap Male \sqcap (\exists Married.Pilote) \sqcap (\forall hasFrere.(Pilote \sqcup Ingenieur)).$$

Le nom *logique de description* vient du fait que ce formalisme est basé sur la logique du premier ordre et que chaque concept plus complexe peut être construit comme une forme de description qui utilise des concepts et des rôles plus simples, comme vu dans l'exemple précédent.

La nécessité de représenter l'information d'une manière structurée et hiérarchisée a donné naissance aux ontologies informatiques, qui seront présentées dans la

section suivante.

6.3 Ontologies informatiques

Le terme *ontologie* vient de la philosophie, qui la définit comme une systématisation de la réalité, d'entités et d'événements. C'est donc la catégorisation de choses existantes et les relations entre ces catégories (Antoniou et van Harmelen, 2004).

En informatique, on s'intéresse à la structure et la nature des choses possibles à représenter, dans notre cas, sous forme d'ontologie. Les ontologies computationnelles sont, au sens large, une forme de modélisation formelle de la structure d'un système, c'est-à-dire de ses entités et de ses relations pertinentes, qu'on obtient par observation (Staab et Studer, 2009).

Du point de vue de la logique, les informations contenues dans une ontologie forment un ensemble d'axiomes qui subsument des relations entre des classes et des propriétés (Baader *et al.*, 2009). Ces relations sont des relations conceptuelles taxonomiques, présentées à la sous-section 1.2.2.

Dans ce contexte, il est important de mentionner *Gros Tas de Notions* (GNT) (Tremblay et Polguère, 2014), une ontologie des concepts centraux de la TST. Cette ontologie a servi de base au développement d'un cours de didactique du lexique français.

Selon Gruber (1995), « une ontologie est une spécification explicite d'une conceptualisation ». Pour Kavalec et Svátek (2005), une ontologie doit formaliser les aspects intentionnels d'un domaine tandis qu'une base de connaissances doit fournir la partie extensionnelle et contenir des affirmations sur des instances de concepts et les relations entre elles. En plus, elle doit être spécifiée d'une manière qui soit compréhensible pour les êtres humains et pour les machines.

Pour accomplir ces objectifs, des langages ont été développés, comme OWL et RDF. Par exemple, OWL est un langage utilisé pour la création et l'instanciation d'ontologies sur le Web. Il est un outil important pour la définition de classes et de relations entre classes (Antoniou et van Harmelen, 2004).

Les langages RDF, RDFS et OWL sont les langages standards du Web sémantique. Ils sont appuyés par le *World Wide Web Consortium* (W3C)[1]. Cependant, d'autres langages ont été développés pour la construction des bases de connaissances. Parmi ces langages, citons (Marchetti *et al.*, 2008) :

— CycL[2] (Lenat et Guha, 1991) : langage déclarative basé sur la logique de premier ordre ;

— F-Logic[3] (Kifer et Lausen, 1989) : langage pour la représentation des connaissances basé sur la même idée de l'orientation à objets des langages de programmation comme Java et C++ ;

— LOOM[4] (MacGregor et Bates, 1987) : langage déclaratif dont la syntaxe est similaire à celle du langage LISP. LOOM est utilisé dans la construction d'une base de connaissance formée par de règles, définitions, faits et un moteur de déduction logique ;

— KIF[5] (Genesereth *et al.*, 1992) : langage basé sur la logique de premier ordre, crée avec l'objectif d'être un langage de communication entre différentes architectures informatiques ;

---

1. https ://www.w3.org/
2. http ://www.cyc.com/
3. https ://www.w3.org/2005/rules/wg/wiki/F-logic
4. http ://www.isi.edu/isd/LOOM/LOOM-HOME.html
5. http ://logic.stanford.edu/kif/specification.html

— Ontolingua [6] (Gruber, 1992) : langage basé sur KIF. Il combine le paradigme des cadres avec la logique du premier ordre. Développé pour la création et l'édition d'ontologies.

Dans les sous-sections suivantes, nous présentons les formalismes créés pour la construction des ontologies dans le Web sémantique : RDF, RDFS, OWL et le langage de requête SPARQL.

6.3.1 Langage RDF

Le *Resource Description Framework* (RDF) est un langage de métadonnées ayant une syntaxe basée sur le XML, pour la modélisation des ressources et des relations entre ressources sur le Web. Une ressource peut être la représentation d'une personne, d'un pays, d'une compagnie, d'un produit, etc. En RDF, les propriétés sont aussi des ressources. Une propriété est une relation entre ressources, comme *âge*, *produit par*, *fils de*, etc.

Un énoncé RDF est un triplet formé par une *ressource*, une *propriété* (prédicat) et une *valeur*. La valeur peut être une ressource ou une constante, comme un numéro ou une chaîne de caractères. On peut considérer un triplet (*ressource*, *propriété*, *valeur*) comme étant une formule du calcul de prédicat : *propriété*(*ressource*, *valeur*). En RDF, il n'y a que des prédicats binaires.

La figure 6.1 montre un exemple d'encodage en RDF de l'affirmation « La capitale du Canada est Ottawa ».

Dans la figure 6.1, « xmlns :rdf » définit la syntaxe RDF, « xmlns :mydomain » informe le local où les définitions des propriétés et des ressources se trouvent, « rdf :Description rdf :about » fait référence à une ressource précédemment définie

---

6. http ://www.ksl.stanford.edu/software/ontolingua/

```
<?xml version="1.0" encoding="UTF-16"?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:mydomain="http://www.mydomain.org/my-rdf-ns">
<rdf:Description rdf:about="Canada">
  <mydomain:capitale>
    Ottawa
  </mydomain:capitale>
</rdf:Description>
</rdf:RDF>
```

Figure 6.1: Code RDF représentant l'affirmation « La capitale du Canada est Ottawa »

de façon unique et « mydomain :capitale » est une propriété définie par l'utilisateur. *Ottawa* est la *valeur*, qui doit aussi être définie comme une ressource.

Dans cette thèse, nous utilisons la syntaxe *Turtle*[7], qui permet d'écrire des triplets RDF d'une façon plus compacte. La même information montrée par la figure 6.1 est montrée par la figure 6.2 en utilisant la syntaxe *Turtle*.

```
@prefix rdf : <http://www.w3.org/1999/02/22-rdf-syntax-ns#">.
@prefix md : <http://www.mydomain.org/my-rdf-ns">.

<#Canada>
  a md:Pays;
  md:capitale <#Ottawa>.
```

Figure 6.2: Code RDF représentant l'affirmation « La capitale du Canada est Ottawa » en utilisant la syntaxe *Turtle*

Dans l'exemple montré par la figure 6.2, « *md* » fait référence à un ensemble de ressources et propriétés définies par l'utilisateur, « *a* » est l'équivalent de « *rdf :about* » et « *a md :Pays* » est l'équivalent de « Le Canada est un pays », et la dernière ligne informe que la capitale du *Canada* est *Ottawa*.

RDF permet aussi la création de vocabulaires. Un vocabulaire RDF est un ensemble de prédicats définis dans un fichier *ontology* qui contient des prédicats d'un

7. https ://www.w3.org/TR/turtle/

domaine. L'avantage de créer des vocabulaires est la possibilité de définir des prédicats qui seront utilisés dans d'autres applications et ontologies, d'une manière analogue à une *Application Programming Interface* (API) en programmation.

Le vocabulaire RDF le plus connu est *Friend of a Friend* (FOAF)[8]. FOAF définit des prédicats représentants les informations les plus communes et générales sur les personnes (*name*, *age*, *familyName*, etc.) et les relations entre personnes (*knows*, etc.) et entre personnes et organisations (*member*, *currentProject*, etc.).

6.3.2 Langage RDF Schéma

RDF Schéma (RDFS) est un langage pour la définition de classes et de hiérarchies de classes. Par exemple, la figure 6.3 montre comment on peut utiliser la propriété *rdfs :subClassOf* pour définir que *Province* fait partie de la classe *Pays*, définie précédemment. À partir de cette figure, les définitions du *prefix rdf* et du *prefix md* seront omises pour alléger les codes.

```
md:Province
  a rdf:Property;
  a rdfs:Class;
  rdfs:subClassOf md:Pays.
```

Figure 6.3: Code RDF pour définir que *Province* fait partie de la classe *Pays*

6.3.3 Langage de requêtes SPARQL

Le langage *Sparql Protocol and RDF Query Language* (SPARQL) est utilisé pour la construction des requêtes sur les données en format RDF. La base de données est vue comme un ensemble de triplets *sujet-prédicat-objet*.

En comparaison à une base de données relationnelle, on peut considérer tous les triplets d'un *sujet* comme étant une seule ligne dans un tableau. Le *sujet* est la clé

8. http ://xmlns.com/foaf/spec/

primaire, chaque *prédicat* est une colonne et les *objets* sont les valeurs de chaque cellule (de chaque combinaison sujet-prédicat). Par exemple, la figure 6.4 montre comment on peut instancier le *sujet* <#*Canada*>.

```
<#Canada>
  a md:Pays;
  md:capitale <#Ottawa>;
  md:population #36000000;
  md:continent <#Amérique>.
```

Figure 6.4: Code RDF pour instancier certaines propriétés de la ressource <Canada>

Les données en format RDF montrées par la figure 6.4 peuvent être vues comme une ligne dans un tableau d'une base de données relationnelle. La clé primaire est *Canada*, les noms des colonnes sont *capital*, *population* et *continent* et les valeurs sont *Ottawa*, *36000000* et *Amérique*.

Une requête en SPARQL a le format suivant :

```
SELECT ?capitale ?pays
WHERE {
    ?x md:nomVille ?capitale ;
       md:estCapitaleDe ?y .
    ?y md:nomPays ?pays ;
       md:continente md:Amérique .
}
```

Cette requête retourne le nom des pays dont le continent de localisation est *Amérique* et le nom de leurs capitales.

La requête suivante va retourner les noms des pays dont la population est plus grande que 15 millions d'habitants :

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX type: <http://dbpedia.org/class/yago/>
PREFIX prop: <http://dbpedia.org/property/>
SELECT ?country_name ?population
```

```
WHERE {
    ?country a type:LandlockedCountries ;
             rdfs:label ?country_name ;
             prop:populationEstimate ?population .
    FILTER (?population > 15000000) .
}
```

Les informations sur les pays sont stockées dans l'ontologie Yago, qui sera présentée plus en détail dans la section 6.4.

6.3.4 Langage OWL

Le langage *Web Ontology Language* (OWL) est un langage créé pour la définition/description d'ontologies, construit sur d'autres langages descriptifs (RDF et RDFS). OWL est basé sur la logique de description (*Description Logic* (DL)), présentée dans la sous-section 6.2.1.

OWL peut être utilisé pour la définition des classes (qui représente des catégories, des entités) et des propriétés (caractéristiques de classes, relations entre classes) ainsi que pour créer de nouvelles descriptions de classes en utilisant des combinaisons d'autres classes (intersections, unions, compléments). Il est aussi possible de définir des contraintes de cardinalité sur les propriétés. Par exemple, une classe déterminée ne peut avoir qu'une seule valeur d'une propriété déterminée.

OWL implémente les constructeurs logiques montrés par le tableau 6.1

Une caractéristique importante d'OWL est qu'il est basé sur l'architecture du Web : il est ouvert (sans propriétaire), utilise le *Internationalized Resource Identifier* (IRI) pour identifier des ressources sur le Web, permet de lier des termes entre ontologies différentes et utilise une syntaxe JSON (*JavaScript Object Notation*)[9] pour faciliter l'échange de données et d'informations.

---

9. https ://www.w3schools.com/js/js_json.asp

Tableau 6.1: Constructeurs logiques implémentés par OWL

allValuesFrom	*complementOf*	*hasValue*
inverseOf	*intersectionOf*	*maxCardinality*
minCardinality	*oneOf*	*someValuesFrom*
unionOf		

L'utilisation des IRI permet aussi d'établir un lien entre toutes les représentations d'une même entité. Par exemple, on peut définir un IRI unique pour l'entité *Canada*, qui sera utilisée pour toutes les applications et les représentations ontologiques sur le Web.

Cette forme de linkage en utilisant des IRI est une forme d'ancrage des symboles (Harnad, 1990). Cependant, elle n'est pas un vrai ancrage des symboles, dans le sens où un symbole (le mot *Canada*) est ancré dans le monde externe. On a plutôt un lien entre un symbole (le mot *Canada*) et un autre symbole, un code alphanumérique (un IRI).

Puisque OWL est basé sur la logique de description, qui est, elle-même, un sous-ensemble de la logique de premier ordre, il est possible d'écrire des raisonneurs pour tester la validité logique d'une ontologie écrite en utilisant OWL, c'est-à-dire, vérifier s'il y a des contradictions parmi les axiomes ontologiques et les relations entre ressources.

Comme d'autres langages de construction d'ontologies, OWL permet aux les utilisateurs d'écrire des conceptualisations explicites et formelles des modèles d'un domaine. Les exigences principales pour atteindre ces objectifs sont (Antoniou et van Harmelen, 2004) :

— Une syntaxe bien définie ;

— Une sémantique bien définie ;

— Un appui efficace pour le raisonnement ;

— Un pouvoir d'expression suffisant ;

— Commodité d'expression.

Une syntaxe bien définie est une nécessité évidente pour n'importe quel langage informatique. Et la syntaxe d'OWL, basée sur JSON ou Turtle, est simple et bien définie. Et, même si le XML n'est pas le langage le plus facile à lire, il y a un certain niveau de commodité à écrire dans ce format.

Une sémantique formelle doit décrire précisément le sens de la connaissance représentée. Précisément signifie que la sémantique est objective. Par exemple, si la sémantique est correcte et précise, il est possible de faire des inférences logiques correctes : si x est membre d'une classe A et A est une sous-classe de B, il est possible d'inférer que x est membre de B.

L'appui pour le raisonnement est une conséquence d'une sémantique bien définie, comme dans l'exemple précédent.

Tel que mentionné auparavant, une ontologie comporte un ensemble d'axiomes. Les axiomes supportés par OWL sont montrés dans le tableau 6.2 (Baader *et al.*, 2009) :

Par rapport au pouvoir d'expression, OWL est une extension de RDF. Avec RDF, essentiellement, il est possible d'exprimer des relations *is-a* (ex. « un chien est un animal »). Avec OWL, il est possible d'exprimer, par exemple, l'union ou la disjonction d'ensembles. Le problème, ici, est que plus un langage est expressif, plus difficile est sa calculabilité. Le sous-ensemble le plus complet d'OWL, OWL-Full, n'est pas calculable. Pour cette raison, d'autres sous-ensembles, moins puissants,

Tableau 6.2: Axiomes supportés par OWL

di erentFrom	*disjointWith*
equivalentClass	*equivalentProperty*
FunctionalProperty	*InverseFunctionalProperty*
sameAs	*subClassOf*
subPropertyOf	*SymmetricProperty*
TransitiveProperty	

sont davantage utilisés, comme OWL-Light et OWL-DL. OWL-DL est la version la plus simplifiée d'OWL encore compatible avec la logique de description.

Les règles à suivre dans la construction d'une ontologie pour qu'elle soit une ontologie DL sont définies par W3C [10].

6.4 Ontologies de haut niveau

Les langages RDF/RDFS et OWL ont été créés et utilisés pour la construction d'ontologies de domaines spécifiques, comme le domaine financier, de l'environnement et de la santé. Par exemple, une des ontologies les plus connues et les plus utilisées est GeoNames [11], qui normalise la nomenclature des entités géographiques (pays, villes, montagnes) et des relations entre ces entités sur le Web.

Une autre ontologie importante a été développée par le département d'informatique de l'Institut Max Planck. Elle est appelée *Yet Another Great Ontology*

10. https ://www.w3.org/TR/2004/REC-owl-ref-20040210/#app-DLinRDF

11. http ://www.geonames.org/

(YAGO) [12].

YAGO est une combinaison de trois autres ontologies : DBPedia [13], qui est une version de Wikipédia [14] en format d'ontologie ; WordNet en format RDF/OWL et GeoNames.

Cependant, pour qu'il y ait une normalisation encore plus large, des ontologies de haut niveau ont été conçues pour la représentation d'entités et de concepts plus généraux, qui sont pertinents à n'importe quel domaine, comme la notion de *concept*, *relation*, *propriété*, etc. Nous présentons ensuite deux de ces ontologies, SKOS et SUMO.

6.4.1 SKOS

Le développement du *Simple Knowledge Organization System* (SKOS) a commencé en 2004. En 2006 [15] le W3C a créé un groupe de recherche pour travailler à son amélioration, pour qu'il puisse devenir un standard de la représentation des connaissances sur le Web. Une première version a été publiée en 2008 [16] et SKOS est devenue un standard en 2009 [17].

La classe de plus haut niveau en SKOS est *Concept*, qui est définie comme une sous-classe de la classe *owl :Class* :

---

12. http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/

13. http ://wiki.dbpedia.org/

14. https ://www.wikipedia.org/

15. https ://www.w3.org/2006/07/swdwg-charter

16. https ://www.w3.org/TR/2008/WD-skos-primer-20080221/

17. https ://www.w3.org/TR/skos-reference/

```
skos:Concept is an instance of owl:Class
```

La motivation pour atteindre cette classe est que, dans chaque ontologie développée, tous les concepts soient définis comme une instance de la classe *skos :Concept*, afin qu'il y ait une compatibilité entre les diverses ontologies. En même temps, chaque développeur d'ontologie est libre de définir ce qu'il considère comme un concept dans son ontologie. Par exemple, on pourrait définir le concept *Lexical-Function* comme suit :

```
lfrep:LexicalFunction rdf:type skos:Concept
```

SKOS utilise l'idée de schéma de concepts pour créer un graphe conceptuel en format d'ontologie. Pour accomplir cela, il y a la classe *skos :ConceptScheme*.

On peut, par exemple, construire un schéma pour représenter les fonctions lexicales du type *skos :ConceptScheme* et définir le concept *fonction lexicale* comme le concept de niveau le plus élevé du schéma. Ensuite, on peut définir le concept *fonction lexicale simple* comme un concept dans notre schéma. Le code serait le suivant :

```
LexicalFunctionsScheme rdf:type skos:ConceptScheme ;
    skos:hasTopConcept lfrep:LexicalFunction .
    lfrep:LexicalFunction skos:topConceptOf LexicalFunctionsScheme .
    lfrep:SimpleLexicalFuntion skos:inScheme LexicalFunctionsScheme .
```

Une autre caractéristique importante de SKOS est la possibilité d'exprimer les relations entre concepts. On connecte deux concepts C_1 et C_2 avec les propriétés *skos :broader*, *skos :narrower* et *skos :related* pour informer que C_1 est un concept plus large, plus étroit ou lié au concept C_2, respectivement. Il faut noter qu'avec SKOS on reste toujours au haut niveau. C'est à chaque ontologie d'implémenter des relations plus fines.

Un dernier point à souligner sur SKOS est la possibilité de créer des collections, en utilisant les classes *skos :Collection* et *skos :OrderedCollection* et les propriétés *skos :member* and *skos :memberList*.

6.4.2 SUMO

Le *Suggested Upper Merged Ontology* (SUMO) (Pease *et al.*, 2002) est une fusion d'autres ontologies de haut niveau construites auparavant.

Contrairement à d'autres ontologies présentées dans cette thèse, SUMO n'est pas construite avec le langage OWL, mais avec le langage *Standard Upper Ontology - Knowledge Interface Format* (SUO-KIF).

SUO-KIF est une implémentation du langage *Knowledge Interchange Format* (KIF), qui est aussi, comme OWL, un langage basé sur la logique de description. La syntaxe de SUO-KIF est similaire à celle du langage fonctionnel LISP.

Nous donnons ci-dessous des exemples de vérités ontologiques écrites en SUO-KIF :

```
(subclass Person Animal)

(and
    (instance JustinTrudeau Human)
    (occupiesPosition JustinTrudeau PrimeMinister Canada))
```

La classe de plus haut niveau en SUMO est la classe *entity*. Cette classe est divisée en deux sous-classes, *physical* et *abstract*. La classe *physical* est divisée en *object* et *process*, et ainsi de suite.

Même si SUMO n'est pas basée sur OWL, il est possible de connecter les ontologies écrites en SUMO avec des ontologies écrites en OWL, car il existe une version de

SUMO traduite vers OWL [18].

Finalement, on souligne qu'en 2008 SUMO a été combinée à YAGO pour la création de l'ontologie Yago-SUMO [19].

6.5 Ontologies métalinguistiques

La conceptualisation ontologique, la sémantique et la lexicologie sont des disciplines fortement connectées. Par exemple, Eluerd (2000) fait les remarques suivantes :

> Par nature, et au de-là de la diversité des écoles, la sémantique est généralisation, conceptualisation. Ainsi, sa description du lexique selon la hiérarchie des catégories ontologiques aristotéliciennes (genre prochain, différences spécifiques) ou selon des relations structurelles (synonymie, antonymie, hyperonymie, hyponymie) relève d'une logique indispensable à ses propres travaux comme à de nombreux aspects des travaux de la lexicologie (Eluerd, 2000, p. 30).

Par rapport à l'information linguistique, on ne s'intéresse pas seulement à stocker des mots ou des expressions isolées, mais aussi à des relations entre mots et expressions, comme les relations de synonymie, hyperonymie et hyponymie, en utilisant les concepts de classe et de sous-classe pour les représenter.

Par exemple, WordNet peut être vue comme une ontologie linguistique ou lexicale, car elle représente des mots et des relations entre mots, même si elle n'est pas organisée comme une vraie taxonomie ontologique.

Cependant, pour la représentation des informations linguistiques en format d'onto-

---

18. http ://www.adampease.org/OP/SUMO.owl

19. http ://people.mpi-inf.mpg.de/ gdemelo/yagosumo/

logie, il faut développer des ontologies métalinguistiques, c'est-à-dire non pas une ontologie qui représente simplement des relations entre mots, mais une ontologie qui représente des concepts linguistiques, comme la « partie du discours », les notions d' « entrée lexicale » et de « sens », des informations sémantiques et syntaxiques, etc.

Dans la sous-section suivante, nous faisons un résumé de l'évolution des principales ontologies métalinguistiques : ISOCat, LMF et LexInfo. Ensuite, dans la section 6.5.2, nous présentons avec plus de détails *lemon*, qui est l'évolution de ces modèles précédents.

6.5.1 Premières ontologies métalinguistiques

L'*ISO TC37 Data Category Registry* (ISOCat) [20] a été la première de ces ontologies métalinguistiques à apparaître.

ISOCat a été proposée et développée par le département de psycholinguistique de l'Institut Max Planck [21].

ISOCat définit des catégories grammaticales, tel que *part of speech*, *syntactic head*, *predicate*, *reflexive verb*, etc.

Le *Lexical Markup Framework* (LMF) [22] (Francopoulo *et al.*, 2006) est un modèle métalinguistique utilisé dans la création de ressources lexicales, telles que les dictionnaires, thésaurus, etc. LMF est devenue un standard ISO en 2008 [23].

---

20. http ://www.isocat.org/

21. http ://www.mpi.nl/

22. http ://www.lexicalmarkupframework.org/

23. https ://www.iso.org/standard/37327.html

Différemment d'ISOCat, dont la fonction est la représentation de catégories grammaticales, le principal objectif de LMF est la construction de ressources lexicales.

LMF utilise ISOCat pour représenter des catégories grammaticales et implémente des classes pour représenter des unités lexicales.

La classe principale dans LMF est *LexicalEntry*, qui représente une entrée lexicale. Cette classe est connectée, par exemple, à des classes : *Lexicon*, *EntryRelation*, *Form*, *Sense*, etc. Il y a encore des classes pour représenter des informations syntaxiques, sémantiques et morphologiques.

LexInfo[24] (Cimiano *et al.*, 2011) est la combinaison de trois modèles précédents :

— LMF.

— LingInfo[25] : implémente des classes lexicales d'une façon similaire à LMF. Il y a, par exemple, les classes *Phrase*, *WordForm*, *InflectedWordForm*, etc.

— LexOnto (Cimiano *et al.*, 2007) : l'objectif principal de LexOnto est l'implémentation des classes nécessaires à la connexion entre ressources lexicales et ontologies d'autres domaines.

LexInfo a combiné les implémentations de ces trois modèles, supprimé les redondances et étendu les possibilités des connexions avec des ontologies d'autres domaines.

Finalement, le *Lexicon model for ontologies* (*lemon*)[26] est une amélioration et extension de LexInfo et de LMF. Le rapport W3C final de la création de *lemon* (qui

24. http ://lexinfo.net/

25. http ://olp.dfki.de/LingInfo/index.php

26. http ://lemon-model.net/

s'écrit toujours en minuscule) a été publié en 2006, comme une recommandation pour la représentation des informations lexicales sur le Web sémantique [27].

Nous présentons *lemon* plus en détail dans la sous-section suivante.

6.5.2 Modèle lemon

lemon (McCrae *et al.*, 2011) est un formalisme du W3C créé pour la représentation de l'information linguistique sur le Web sémantique.

Les créateurs de *lemon* le décrivent de la manière suivante [28] :

> The aim of the lexicon model for ontologies (*lemon*) is to provide rich linguistic grounding for ontologies. Rich linguistic grounding includes the representation of morphological and syntactic properties of lexical entries as well as the syntax-semantics interface, i.e. the meaning of these lexical entries with respect to an ontology or vocabulary (www.w3.org/2016/05/ontolex/).

L'ontologie *lemon* est composée de cinq modules :

- Le module *Core*, représenté par le préfixe *ontolex*, illustré par la figure 6.5.
- Le module *Syntax and Semantics* (*synsem*), montré par la figure 6.6.
- Le module *Decomposition* (*decomp*), pour la représentation des EPL comme des composants d'un objet *LexicalEntry*.
- Le module *Variation & Translation* (*vartrans*), pour la représentation de différentes formes d'un mot (par exemple, des variations entre l'anglais britannique et l'anglais américain) et pour faire la connexion des différents mots en différentes langues, à travers une classe *LexicalConcept*.

---

27. https ://www.w3.org/2016/05/ontolex/

28. https ://www.w3.org/2016/05/ontolex/

— Le module *Linguistic Metadata* (*lime*), pour représenter les métadonnées au niveau de l'interface lexico-ontologique.

La figure 6.5 montre le modèle *core* de *lemon*. Il faut noter qu'il y a une classe pour la représentation des EPL.

Les ontologies lexicales construites en suivant le format *lemon* peuvent se connecter à des ontologies externes (qui peuvent représenter n'importe quel domaine, comme l'économie, le sport, la politique, etc.) par la classe *LexicalEntry* ou par la classe *LexicalSense*.

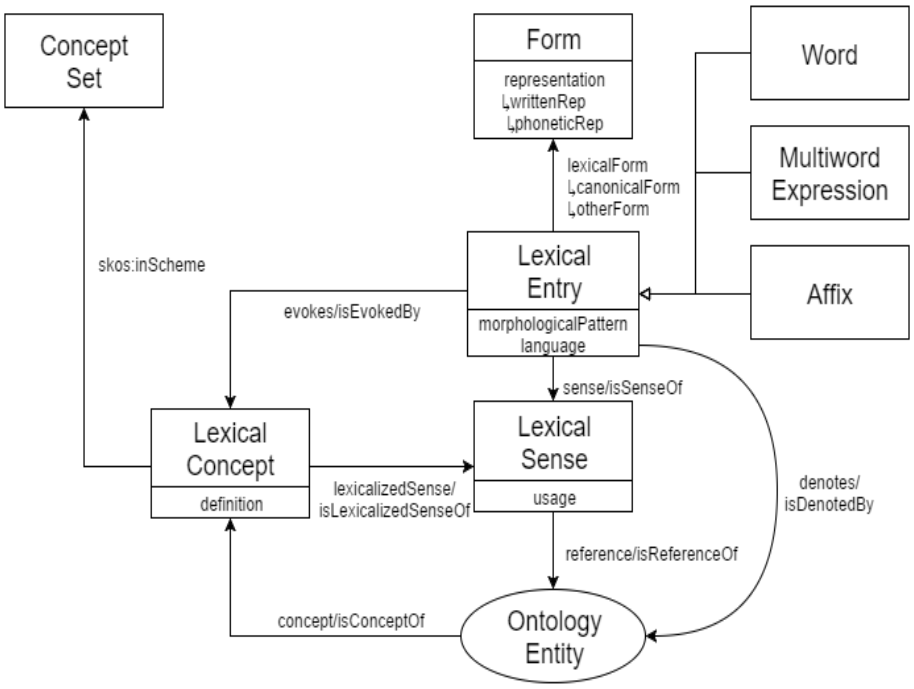

Figure 6.5: Modèle *core* de *lemon* (www.w3.org/2016/05/ontolex/)

Parmi les principaux avantages de *lemon* par rapport aux modèles précédents, comme LexInfo et LMF, citons :

— la séparation entre l'information linguistique et l'information ontologique.

— des informations linguistiques, telles que *partOfSpeech* et *writenForm* sont représentées comme des propriétés RDF, différemment de LMF, qui les représente comme des attributs d'une propriété. Cela facilite, par exemple, l'utilisation du langage de requête SPARQL.

— *lemon* utilise ISOCat pour représenter des propriétés comme *partOfSpeech*, *gender* and *tense*.

— il est facile d'étendre *lemon* et créer de nouvelles formes de représentation, comme la représentation des fonctions lexicales, développée dans notre modèle.

— il y a déjà plusieurs ressources lexicales en format *lemon*, comme WordNet (Fellbaum, 1998), Dbnary [29], FrameBase [30] et DBPedia Wiktionary [31].

Dans *lemon*, les mots sont représentés par les classes *LexicalEntry* et *LexicalForm*. La classe *LexicalEntry* est connectée à une classe *LexicalSense*, qui représente le sens d'un mot, une unité lexicale. Cela permet de séparer la forme et le sens d'un mot (Lyons, 1995), comme mentionné dans la section 1.5.

La connexion entre un objet *lemon* et des ontologies externes est faite par la classe *LexicalSense* ou par la classe *LexicalEntry*.

La figure 6.6 présente le module *Syntax et Semantics* (synsem), lequel peut être utilisé pour représenter la syntaxique d'un mot représenté par une classe *LexicalEntry*.

Comme exemple de l'application du module *synsem*, la figure 6.7 montre la repré-

29. http://kaiko.getalp.org/about-dbnary/development/

30. http://www.framebase.org/data

31. http://dbpedia.org/Wiktionary

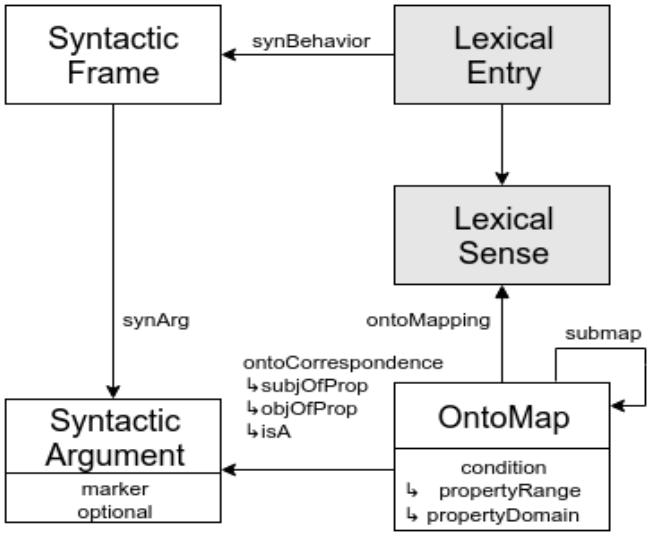

Figure 6.6: Module *synsem* de *lemon* (www.w3.org/2016/05/ontolex/)

sentation du comportement syntaxique du verbe anglais *to own* (posséder) et la figure 6.8 montre la même information en format RDF.

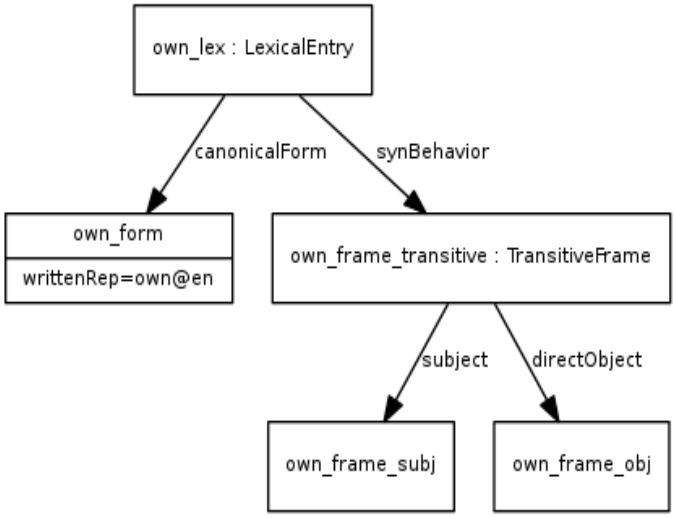

Figure 6.7: Représentation schématique du comportement syntaxique du verbe *to own* (www.w3.org/2016/05/ontolex/)

Dans notre modèle proposé, présenté dans le chapitre 8, la base et la valeur d'une fonction lexicale sont représentées par des classes *lemon LexicalSense*.

```
:own_lex a ontolex:LexicalEntry ;
      ontolex:canonicalForm :own_form ;
      synsem:synBehavior :own_frame_transitive .

:own_form ontolex:writtenRep "own"@en.

:own_frame_transitive a lexinfo:TransitiveFrame;
      lexinfo:subject :own_frame_subj;
      lexinfo:directObject :own_frame_obj.
```

Figure 6.8: Représentation *lemon* du comportement syntaxique du verbe *to own* (www.w3.org/2016/05/ontolex/)

En tenant compte de l'exemple donné dans la sous-section 3.2.5, dans notre modèle, le mot *océan* est représenté par un objet *lemon LexicalEntry* et ses différents sens, $Océan_{I:1a}$, $Océan_{I:1b}$, $Océan_{I:2}$ et $Océan_{II}$ sont chacun représentés par un objet *lemon LexicalSense*. Chaque sens est un lexème du mot *océan* et ils sont représentés par des sous-scripts, tel qu'expliqué dans la sous-section 3.2.5.

La connexion sémantique représentée par une FL est entre des sens et non entre des formes lexicales ou des entrées lexicales. De cette manière, nous pouvons construire un réseau lexical désambiguïsé, en connectant des unités lexicales avec des FL.

Finalement, puisque chaque unité lexicale de notre modèle est également représentée par une classe *LexicalEntry*, on peut utiliser le module *lemon* synsem [32] (Figure 6.6) pour représenter le comportement syntaxique d'une unité lexicale.

6.5.3 Représentation d'expressions polylexicales avec *lemon*

Dans cette sous-section, nous présentons un exemple d'utilisation du module *lemon decomp* [33] pour représenter une EPL.

La figure 6.9 montre la représentation de l'EPL *African swine fever* et la figure

32. https://www.w3.org/2016/05/ontolex/#syntax-and-semantics-synsem

33. https://www.w3.org/2016/05/ontolex/#decomposition-decomp

6.10 montre un code RDF représentant cette EPL.

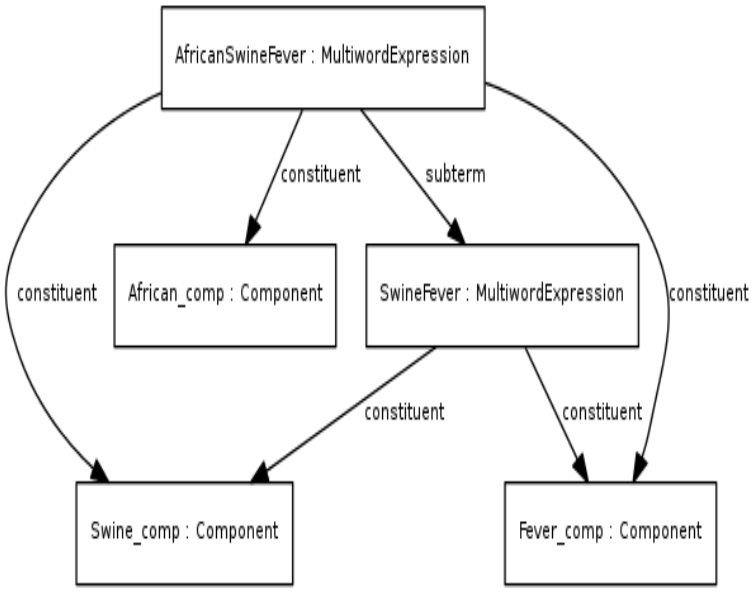

Figure 6.9: Représentation de l'EPL *African swine fever* en utilisant le module *lemon decomp* (www.w3.org/2016/05/ontolex/)

```
:AfricanSwineFever a ontolex:MultiwordExpression;
        decomp:constituent :African_comp, :Swine_comp, :Fever_comp;
        decomp:subterm :SwineFever.
:African_comp a decomp:Component.
:Swine_comp a decomp:Component.
:Fever_comp a decomp:Component.
:SwineFever a ontolex:MultiwordExpression;
        decomp:constituent :Swine_comp, :Fever_comp.
```

Figure 6.10: Code RDF représentant l'EPL *African swine fever* en utilisant le module *lemon decomp* (www.w3.org/2016/05/ontolex/)

Cet exemple montre comment n'importe quelle EPL peut être représentée en utilisant le modèle *lemon*. La représentation de l'EPL elle-même, en utilisant le module *decomp*, est superficielle. Cependant, elle peut être combinée avec la représentation de sa structure de phrase syntaxique pour rendre compte des particularités syntaxiques d'une EPL.

De plus, chaque mot qui fait partie d'une EPL est représenté par un objet *Lexica-*

lEntry, auquel d'autres informations syntaxiques, sémantiques et morphologiques peuvent être connectées.

La figure 6.11 montre le code RDF représentant la structure de phrase syntaxique de cette EPL. Elle utilise le vocabulaire OLiA [34] et le tagset Penn TreeBank [35].

```
:AfricanSwineFever_root a decomp:Component ;
     decomp:correspondsTo :AfricanSwineFever ;
     decomp:constituent :African_node, :SwineFever_node ;
     rdf:_1 :African_node;   rdf:_2 :SwineFever_node;
     olia:hasTag penn:NP .
:African_node a decomp:Component ;
     decomp:correspondsTo :African ;
     olia:hasTag penn:JJ .
:SwineFever_node a decomp:Component ;
     decomp:constituent :Swine_node, :Fever_node ;
     rdf:_1 Swine_node; rdf:_2 Fever_node;
     olia:hasTag penn:NP .
:Swine_node a decomp:Component ;
     decomp:correspondsTo :Swine ;
     olia:hasTag penn:NN .
:Fever_node a decomp:Component ;
     decomp:correspondsTo :Fever ;
     olia:hasTag penn:NN .
```

Figure 6.11: Code RDF représentant la structure de phrase syntaxique de l'EPL *African swine fever* (www.w3.org/2016/05/ontolex/)

6.6 Conclusion

Nous avons vu dans ce chapitre la définition et la motivation derrière la création du Web sémantique, ses formalismes (les langages RDF/RDFS/OWL) et la notion d'ontologie informatique. Ces langages sont devenus le premier cas d'utilisation globale d'un formalisme créé pour la représentation des connaissances, tel que l'affirment Hendler et van Harmelen (2008, p. 823) : « The languages RDF, RDFS and OWL are without question the most widely used KR languages in history. »

---

34. http ://nachhalt.sfb632.uni-potsdam.de/owl/

35. https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

Les ontologies informatiques partagées sur le Web sont des outils très importants, car on peut les réutiliser sans nécessairement devoir recréer et rassembler des millions d'informations sur plusieurs domaines.

Les formalismes du Web sémantique ont été développés pour permettre la représentation et le partage des connaissances sur le Web. Des ontologies ont été développées en les utilisant.

Cependant, pour bien représenter des informations sémantiques, syntaxiques et morphologiques, les langages RDF/RDFS/OWL ne sont pas suffisants. Pour cette raison, des ontologies dites métalinguistiques ont été développées.

À partir d'ISOCat, construite pour la représentation de catégories linguistiques, ces ontologies ont évolué jusqu'à *lemon*, le standard W3C pour la représentation des informations linguistiques, y compris la représentation des expressions polylexicales.

Dans le chapitre 8, nous présentons notre modèle de représentation des fonctions lexicales et des collocations, qui utilise *lemon* pour représenter des formes (entrées lexicales) et des sens des mots.

Avant, dans le chapitre 7, nous présentons des travaux qui portent sur la représentation des expressions polylexicales et sur la représentation et l'extraction des collocations, en utilisant les fonctions lexicales et d'autres formalismes. Nous montrons aussi d'autres applications pour les fonctions lexicales, comme la traduction automatique et la génération de texte.

CHAPITRE VII

REPRÉSENTATION D'EXPRESSIONS POLYLEXICALES ET DE COLLOCATIONS

7.1 Introduction

Plusieurs travaux ont été consacrés à l'extraction de collocations et d'expressions polylexicales à partir de textes. Cependant, la représentation de collocations et d'expressions polylexicales est encore un sujet peu abordé (Ramisch, 2012).

Dans ce chapitre, nous présentons l'état de l'art de la représentation et de l'extraction de collocations en utilisant les fonctions lexicales et la représentation d'expressions polylexicales en général. À la section 7.2, nous présentons trois différentes approches pour la représentation des expressions polylexicales en général. À la section 7.3, nous présentons des méthodes pour la représentation de collocations en utilisant les fonctions lexicales.

À la section 7.4, nous faisons le résumé des travaux portant sur la représentation de collocations et d'expressions polylexicales et faisons une comparaison entre ces travaux et notre proposition de représentation de collocations. À la section 7.5, nous présentons des travaux qui portent sur l'extraction des collocations en utilisant les fonctions lexicales. Finalement, à la section 7.6, nous présentons d'autres applications pour les fonctions lexicales, comme la traduction automatique, la génération de texte et la construction des dictionnaires bilingues.

7.2 Représentation d'expressions polylexicales

Dans cette section, nous présentons des travaux sur la représentation d'expressions polylexicales en général, ou de sous-groupes d'expressions polylexicales autres que les collocations. Malgré qu'ils ne portent par directement sur les collocations, ces travaux sont pertinents pour notre thèse, puisqu'ils permettent de dresser un portrait général de la représentation des mots composés, dont les collocations font partie.

7.2.1 Représentation d'expressions idiomatiques et de verbes à particule en deux niveaux

Villavicencio *et al.* (2004) présentent un formalisme pour la représentation d'expressions idiomatiques et de verbes à particule de langue anglaise.

Premièrement, pour leur étude, ils utilisent un encodage pour représenter des mots simples, appelés *simplex*. Chaque simplex représente une combinaison unique sens-partie du discours d'un vocable. Par exemple, il y a un simplex pour le vocable LIKE comme verbe et un simplex pour le vocable LIKE comme conjonction.

Un tableau est utilisé pour représenter l'ensemble des simplex, une ligne par simplex. Chaque simplex est représenté par quatre caractéristiques, une par colonne du tableau : identificateur (unique pour chaque simplex), orthographe, type et prédicat. Le type représente la partie du discours et le prédicat représente la relation normale avec le sujet d'une phrase.

Par exemple, la représentation du vocable LIKE utilisé comme verbe est la suivante :

— identificateur : like_tv_1

— orthographe : like

— type : trans-verb

— prédicat : like_v_rel

Les EPL sont représentées en deux tableaux. Le premier tableau représente le sens idiomatique de chaque simplex et le deuxième tableau représente les EPL.

Le premier tableau contient cinq colonnes :

— identificateur : comme dans le tableau de simplex, mais chaque identificateur commence par *i*, pour sens idiomatique ;

— forme de base : le correspondant dans le tableau de simplex ;

— type : la partie de discours, ajoutée du mot *idiomatic* ;

— prédicat : comme dans le tableau de simplex, préfixé par *i* ;

— paraphrase : représente le sens idiomatique.

Prenons comme exemple l'expression idiomatique anglaise *spill the beans*, qui signifie *révéler le secret*. Le tableau 7.1 représente les simplex de cette expression.

Dans le deuxième tableau, chaque EPL est représentée par deux ou plusieurs lignes, chacune contenant une des composantes d'EPL. Par exemple, le tableau 7.2 représente l'expression *spill the beans*.

Tableau 7.1: Représentation des simplex de l'expression idiomatique *spill the beans*

Identificateur	**Forme de base**	**Type**	**Prédicat**	**Paraphrase**
i_spill_tv_1	spill_tv_1	idiomatic-trans-verb	i_spill_tv_rel	reveal_tv_rel
i_bean_n_1	bean_n_1	idiomatic_noun	i_bean_n_rel	secret_n_rel

Tableau 7.2: Représentation de l'expression idiomatique *spill the beans*

Phrase	Component	Prédicat	Slot	Optionel
i_spill_beans_1	i_spill_tv_1	-	PRED1	no
i_spill_beans_1	i_bean_n_1	-	PRED2	no

7.2.2 Représentation d'expressions polylexicales en utilisant la méthode des classes d'équivalence

Grégoire (2010) présente un formalisme basé sur la méthode des classes d'équivalence (MCE) (Odijk, 2004) pour la représentation des EPL appliquée au néerlandais. Ce formalisme est utilisé pour l'implémentation d'une ressource lexicologique appelée Dutch Electronic Lexicon of Multiword Expressions (DuELME).

L'objectif de la MCE est la représentation des EPL avec des classes d'équivalence (CE). Chaque CE est représentée par un patron de description, formé par la catégorie syntaxique du mot principal des expressions représentées par la CE, les compléments nécessaires, la description de la structure interne des compléments et des informations morpho-syntaxiques de chaque composant. Voici un exemple de patron de description :

— *Expressions dont le mot principal est un verbe qui prend un objet direct formé par un déterminant et un nom singulier*. (1)

Les expressions suivantes suivent ce patron : *prendre l'autobus*, *faire la vaisselle*, *pousser un cri*, etc.

Le problème avec cette approche est qu'elle résulte en un très grand nombre de CE, dont plusieurs sont de petites tailles. Pour régler ce problème, Odijk (2004) propose de paramétriser les classes d'équivalence.

Plusieurs patrons d'expression sont similaires et ne présentent qu'une différence locale minimale. Par exemple, sur cet autre patron :

— *Expressions dont le mot principal est un verbe qui prend un objet direct formé par un déterminant et un nom pluriel*. (2)

L'introduction d'un paramètre du nombre (singulier ou pluriel) permet la fusion des patrons (1) et (2) dans un seul patron :

— *Expressions dont le mot principal est un verbe, qui prend un objet direct formé par un déterminant et un nom* [$param_1$]. - $param_1$ = {singulier, pluriel} (3)

Grégoire (2010) utilise deux types de description : un pour représenter les CE et un pour représenter les EPL. Les CE sont représentées par une description textuelle du patron d'expression et une notation formelle. La notation utilisée pour décrire les patrons est le formalisme d'arbres de dépendances (Grégoire, 2007). Nous pouvons voir ici l'exemple (description textuelle + description formelle) :

— *Expressions dont le mot principal est un verbe qui prend un objet direct formé par un déterminant et un nom.* Par exemple : *faire la vaisselle* – [.VP [.obj1 : NP [.det : D (1)] [.hd :N (2)]][.hd :V (3)]]

Dans l'exemple ci-dessus, les étiquettes de dépendance apparaissent en minuscule (obj1 = objet de la phrase nominale, det = déterminant (article), hd = tête de la phrase) et les étiquettes de catégorie en majuscules (VP = phrase verbale ; NP = phrase nominale ; D = déterminant ; N = nom ; V = verbe).

Les étiquettes de dépendance et les catégories sont séparées par deux points (par ex. : « .obj1 : NP »). Dans cet exemple, nous avons :

— obj1 et NP = *faire la vaisselle*

— det = « *la* »

— tête de la phrase nominale (.hd :N) = *vaisselle*

— tête de la phrase verbale (.hd :N) = *faire*

Il est possible d'avoir le mot *liste* entre parenthèses après l'étiquette de catégorie, ce qui signifie qu'un des membres d'une liste dans le tableau d'EPL peut apparaître dans cette position. Les nœuds qui sont des feuilles sont suivis d'un indice, qui se réfère au composant de l'EPL représenté dans le champ « liste de composants » dans le tableau de description des EPL.

Le tableau de description des EPL contient toutes les EPL représentées par le système DuELME (5000 EPL du néerlandais). Les principaux champs du tableau sont :

— L'acronyme du patron (CE) dont l'EPL fait partie ;

— L'expression polylexicale ;

— La liste de composants : une liste des formes canoniques de chaque mot de l'EPL ;

— Une liste des composants qui peuvent apparaître dans la position indiquée par « (liste) » dans la description formelle de l'EC.

7.2.3 Formalismes graphique et linéaire pour la représentation d'expressions polylexicales

Graliński *et al.* (2010) présentent deux formalismes pour la représentation d'EPL : Multiflex et Poleng.

Multiflex est basé sur l'idée de représentation par graphes en deux couches. Chaque graphe représente une forme possible d'EPL (plusieurs EPL suivent un même

patron morpho-syntaxique, représenté par un seul type de graphe). Chaque graphe est formé de deux couches. La première représente les possibles composants, avec les cas, genre et nombre (pluriel ou singulier) possibles.

La deuxième couche représente une variation possible de l'ordre des mots dans la première couche (par ex. : le mot qui apparaît dans la première position dans la couche *1*, avec le cas nominatif, peut apparaître dans la position *3*, avec le cas accusatif.

Une description entre les couches indique de quel composant les EPL suivant ce graphe prennent leurs formes (par ex. : chaque mot d'EPL doit prendre le cas du deuxième composant et le genre et nombre du premier composant).

Le formalisme utilisé par Poleng est une chaîne de caractères linéaire, plus concise que Multiflex. Par exemple, l'expression *black co ee* serait représentée comme suit :

— N : 3s [black_A coffee_N !]

- le premier N : l'expression est un nom ;
- 3 : le genre de l'expression est neutre ;
- s : l'expression est au singulier ;
- le premier mot de l'expression est un adjectif (A) et le deuxième est un nom (N)
- le symbole ! : indique la tête de l'expression

7.2.4 Représentation des expressions polylexicales dans XMG

Lichte *et al.* (2016) présentent une approche générale orientée objet pour représenter les EPL, basée sur le *eXtensible Meta Grammar* (XMG) (Crabbé *et al.*, 2013). XMG stipule des langages de description qui sont organisés en classes.

Par exemple, la classe *intransitive*[ ] est utilisée pour encoder des verbes intransitifs. Cette classe contient deux autres classes, *subject*[ ] et *verb*[ ], et utilise un opérateur de priorité pour encoder la relation syntaxique entre un sujet et un verbe. La classe *transitive*[ ] importe la classe *intransitive*[ ] et la classe *object*[ ], et définit les relations syntaxiques entre le sujet et l'objet et entre l'objet et le verbe.

Chaque EPL individuelle est encodée comme une classe. Par exemple, l'EPL *to seize the opportunity* peut être encodée comme une classe qui importe la classe *transitive*[ ], informe des variations morphologiques possibles (nombre, personne, etc.) du sujet et de l'objet comme des paramètres variables et fixe la variable *verb*, ici *seize*.

7.3 Représentation des collocations en utilisant les fonctions lexicales

Dans cette section, nous présentons des travaux qui portent sur la représentation des collocations en utilisant les FL.

7.3.1 Perspectives sur l'utilisation des fonctions lexicales

Heid et Raab (1989) sont parmi les premiers à avoir ouvert les perspectives sur l'utilisation des FL pour représenter les collocations.

Leur étude propose que l'information sur la possibilité d'utiliser un lexème dans une collocation soit représentée dans les entrées de dictionnaires des mots qui participent comme des *mots clés* à différentes collocations.

Les auteurs ont établi (pour l'anglais) une relation entre les classes grammaticales qui peuvent apparaître comme mots clés et les classes grammaticales de leurs collocatifs respectifs possibles (pour un exemple, voir le tableau 7.3).

Les contributions les plus importantes de cet article sont ses propositions pour l'identification automatique de collocations en utilisant les FL :

— Analyser comment la catégorie grammaticale du collocatif est prévisible à partir de la fonction et de la base ;

— Identifier le type de FL applicable à un lexème en considérant la classe sémantique du lexème ;

— Généraliser la formation de collocations pour les membres de classes sémantiquement homogènes. Considérons, par exemple, la classe C = {*fichier*, *information*, *message*, *donné*, *mot-de-passe*}. L'application de la FL complexe LiquFunc_0, où *Liqu* est la fonction dont le sens est « causer la fin » et *Func* est la fonction dont le sens est « exister » : LiquFunc_0 (C) = *supprimer* (*supprimer le fichier*, *supprimer l'information*, *supprimer le message*, etc.) ;

— Analyser comment les formes syntaxiques de certains collocatifs peuvent être décrites à partir de règles.

Tableau 7.3: Possibles classes grammaticales des collocatifs selon la base de la collocation

Base	Collocatifs possibles
nom	nom, verbe, adjectif
verbe	adverbe
adjectif	adverbe

7.3.2 Intégration entre les fonctions lexicales et la grammaire HPSG

Heylen *et al.* (1994) font l'intégration du concept de FL avec la grammaire *Head-driven Phrase Structure Grammar* (HPSG) (Pollard et Sag, 1994) pour représenter les collocations.

Ce travail est similaire à celui d'Erbach et Krenn (1993), qui utilise le formalisme HPSG pour représenter des expressions idiomatiques et des verbes supports en langue allemande.

HPSG est une grammaire générative (un ensemble de règles pour la génération de toutes les phrases possibles d'une langue à partir des combinaisons possibles des mots) de type *Phrase Structure Grammar* (PSG) (Chomsky, 1957). Une PSG est une grammaire du type constituant.

Dans HPSG, chaque signe (mot ou phrase) est représenté par une *Head-driven Phrase Structure Grammar* (matrice attribut-valeur) (MAV). Chaque mot a deux attributs : PHON (la forme phonétique du mot) et SYNSEM (l'information syntaxique et sémantique). Le SYNSEM est divisé en sous-attributs. La figure 7.1 montre la MAV du mot anglais *she* (elle).

$$\begin{bmatrix} word & & \\ \text{PHON} & \langle \text{'she'} \rangle & \\ \text{SYNSEM} & \begin{bmatrix} synsem & \\ \text{CAT} & \begin{bmatrix} \text{HEAD} & noun \end{bmatrix} \\ \text{CONT} & \begin{bmatrix} ref\text{-}index & \\ \text{PER} & 3rd \\ \text{NUM} & sing \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Figure 7.1: MAV représentant le mot she (Heylen *et al.*, 1994)

Heylen *et al.* (1994) utilisent cette structure de la MAV et les FL pour représenter les collocations. La figure 7.2 montre comment la collocation *strong criticism* est représentée. Cette collocation est modélisée par la FL *Magn* (intensificateur) : Magn(*criticism*) = {*strong*}.

La collocation est représentée dans l'entrée de la base (*criticism*), où il y a une

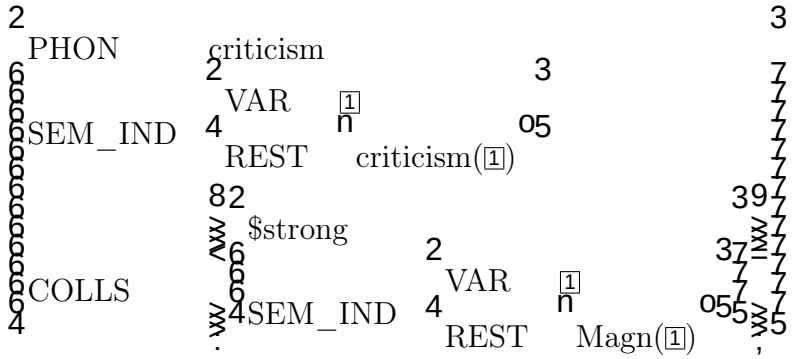

Figure 7.2: MAV représentant la collocation *strong criticism* (Heylen *et al.*, 1994)

zone (*COLLS*) pour représenter toutes les collocations dont cette base fait partie. Dans l'exemple, le collocatif *strong* est montré. Dans son champ sémantique (SEM_IND) il y a l'information que ce collocatif est lié à la base par la FL *Magn*. L'interprétation de cette représentation est exprimée par Heylen *et al.* (1994) de cette manière : « One should read the feature structure as specifying that the semantics of strong (as a collocate) is the predicate *Magn([1])*. »

7.3.3 Représentation de collocations en format XLE

Lareau *et al.* (2012) montrent comment la *glue semantics* (Dalrymple *et al.*, 1993) est utilisée pour importer les FL dans le *Lexical-Functional Grammar* (LFG) (Kaplan et Bresnan, 1982), comment elle est implémentée dans la *Xerox Linguistics Environment* (XLE)[1] (Crouch et King, 2006) et comment leur formalisme est utilisé pour la représentation de collocations.

Ils proposent trois manières de représenter les FL et les collocations dans XLE :

— dans la structure fonctionnelle (*f-structure*) d'un XLE ;

— dans la structure sémantique (*s-structure*) d'un XLE ;

1. http ://www2.parc.com/isl/groups/nltt/xle/doc/xle_toc.html

— comme une implémentation basée sur les transferts.

Nous montrons ici les deux premières formes de représentation des collocations dans XLE, telles que présentées par les auteurs : dans la *f-structure* et dans la *s-structure*.

La *f-structure* (Kaplan et Bresnan, 1982) représente des fonctions grammaticales de surface, des notions comme *sujet*, *objet* et *adjoint*. Lareau *et al.* (2012) proposent le remplacement de collocatifs par les noms des fonctions lexicales dans l'attribut prédicat (PRED) d'une *f-structure*.

La figure 7.3 montre comment la phrase « Mark kicked a beautiful goal » (*Mark a frappé un beau but*) est encodée dans la *f-structure* de XLE. Il y a deux collocations dans cette phrase :

— *kicked a goal*, modélisée par la FL $Oper_1$: $\text{Oper}_1(goal) = \{to\ kick\}$;

— *beautiful goal*, modélisée par la FL *Bon* : $\text{Bon}(goal) = \{beautiful\}$.

$$\begin{bmatrix} & & & \\ \text{PRED} & \text{'Oper1} \left\langle [1:\text{Mark}], [12:\text{goal}] \right\rangle\text{'} & & \\ \text{SUBJ} & \boxed{1}\ \text{PRED 'Mark'} & & \\ \text{OBJ} & \boxed{12}\begin{bmatrix} \text{PRED} & \text{'goal'} \\ \text{ADJUNCT} & \{\ 14\ \text{PRED 'Bon'}\ \} \\ \text{DEF} & - \end{bmatrix} & & \\ \text{TENSE} & \text{PAST} & & \end{bmatrix}$$

Figure 7.3: Représentation de la phrase « Mark kicked a beautiful goal » en utilisant une FL pour représenter les collocations dans la *f-structure* d'un XLE (Lareau *et al.*, 2012)

Une *s-structure* est dérivée d'une *f-structure* pour encoder la sémantique des expressions. C'est, selon Lareau *et al.* (2012, p. 12), un « graphe acyclique direct

connecté qui encode uniquement les relations prédicat-argument ». Les auteurs continuent comme suit :

> The labels of the nodes, given by the attributes SEM in the AVMs, are either the name of a lexeme when it has a literal reading, or the name of a LF when it is a meaningful collocate (e.g., 'Bon' instead of 'good', 'Magn' instead of 'big' or 'intense') (Lareau *et al.*, 2012, p. 12-13).

La figure 7.4 montre une *s-structure* pour la collocation *beautiful goal* à gauche et sa représentation MAV à droite. Dans cet exemple, les lexèmes *goal* et *Mark* sont représentés dans leurs formes littérales lorsque le lexème *beautiful* est remplacé par son sens idéalisé, représenté par la FL *Bon*.

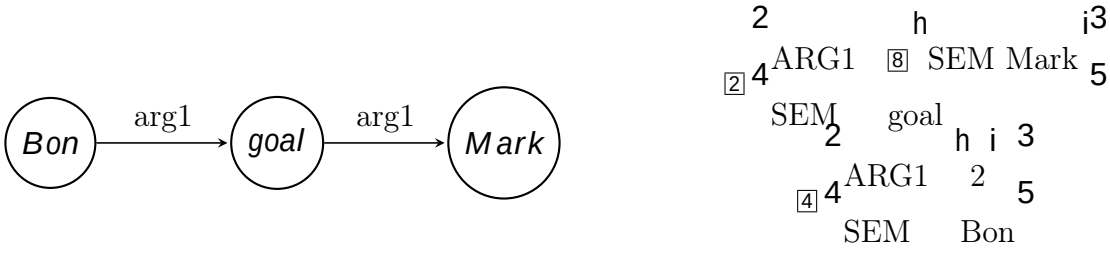

Figure 7.4: Représentation de la collocation *beautiful goal*, de la phrase « Mark kicked a beautiful goal », en utilisant la FL *Bon* pour remplacer un collocatif dans la *s-structure* d'un XLE (Lareau *et al.*, 2012)

De façon similaire, Lambrey (2016) implémente plus de 26 000 patrons de FL pour un générateur de texte basé sur la grammaire MARQUIS (Lareau et Wanner, 2007; Wanner *et al.*, 2010). MARQUIS établit des règles générales pour l'interface sémantique-syntaxe. Cette grammaire est basée sur MATE (Bohnet et Wanner, 2010), un transducteur de graphes qui modélise les niveaux de modélisation structurale dans la TST, tel que montré à la section 3.3. MATE facilite la transformation entre plusieurs graphes, par exemple entre un graphe sémantique et un graphe syntaxique profond.

7.3.4 Ontologie lexicale utilisant les formalismes du Web sémantique

Lefrançois et Gandon (2012) présentent une ontologie lexicale basée sur la théorie Sens-Texte et qui utilise les langages RDF/OWL pour la construction d'un dictionnaire. Leur approche est basée sur une architecture en trois couches :

— la couche méta-ontologique ;

— la couche ontologique ;

— la couche de données.

La couche méta-ontologique est formée par les méta-classes qui seront instanciées dans la couche intermédiaire (ontologique). En voici un autre exemple : la méta-classe *ILexicalUnit* représente tous les types d'unités lexicales, par exemple *forme* et *vivant*. Dans la couche intermédiaire, une classe *Entity* est connectée à la méta-classe *ILexicalUnit* par une relation *is-a*. Un autre exemple est la méta-classe *ISemanticRelation*, qui est une super-classe des relations sémantiques représentées dans la couche intermédiaire.

La couche ontologique est la couche intermédiaire du modèle. Cette couche est formée par des classes qui représentent des concepts, comme *Entity*, *Person* et *State*. Ces classes sont connectées par des relations sémantiques, qui sont des instances de la méta-classe *ISemanticRelation* de la couche méta-ontologique.

La couche de données représente les instances des classes de la couche ontologique. Par exemple, on peut avoir *Mary01* comme instance de la classe *Person* ou *Alive01*, comme instance de la classe *Alive*.

Il y a encore une quatrième couche, au-dessus de la couche méta-ontologique, qui est la couche OWL. Dans cette couche, il y a des classes et des propriétés OWL, comme *owl :Class* et *owl :ObjectProperty*. Les méta-classes de la couche méta-

ontologique sont connectées à ces classes et propriétés. Par exemple, *ILexicalUnit* est connectée à *owl :Class* et *ISemanticRelation* est connectée à *owl :ObjectProperty*.

Les auteurs expliquent que cette division par couches garantit l'accomplissement de trois des quatre principes de rédaction d'un dictionnaire explicatif et combinatoire (Mel'čuk *et al.*, 1995) : les principes de formalité, de cohérence interne et de traitement uniforme. Le principe d'exhaustivité n'est pas accompli.

Dans ce modèle, les collocations sont représentées comme des entrées de dictionnaire du mot clé de la collocation.

7.3.5 Représentation des relations syntagmatiques dans WordNet

Nous présentons dans cette sous-section des travaux qui traitent de la création de relations syntagmatiques dans WordNet, soit ceux de Bentivogli et Pianta (2004), Nica *et al.* (2004) et Schwab *et al.* (2007). Ces travaux traitent non seulement des collocations, mais aussi des relations syntagmatiques en général, et ils n'utilisent pas les FL ou les formalismes du Web sémantique.

Bentivogli et Pianta (2004) proposent pour WordNet la création d'un nœud pour représenter une collocation et la création de relations entre le nouveau nœud et les mots qui les constituent.

Par exemple, pour représenter la collocation *peur bleue*, il faut créer un nœud *peur bleue*, connecté aux synsets qui représentent les mots *peur* et *bleue*, en utilisant des relations *composé de*. Le problème est que, comme en WordNet, les relations sont entre synsets, on devrait avoir dans le synset de *peur* le mot *crainte*, par exemple. Et la connexion entre synsets offrirait la possibilité d'avoir l'expression **crainte bleue*, qui n'existe pas.

Nica *et al.* (2004) proposent l'enrichissement de la partie espagnole d'EuroWordNet avec des patrons syntaxiques annotés avec des sens. Pour chaque nom *X* dans un corpus, ils font l'extraction des expressions selon certains patrons syntaxiques, comme N-ADJ, ADJ-N, N-PREP-N, etc.

Ensuite, pour chaque patron, les noms les plus fréquents selon le même patron sont extraits, pour former un ensemble de noms qui ont des relations paradigmatiques avec *X*. Les mots qui co-occurrent avec *X* dans les patrons forment l'ensemble des mots qui ont des relations syntagmatiques avec *X*.

Des heuristiques de désambiguïsation sont appliquées à ces deux ensembles pour identifier les différents sens possibles de *X*. Un tableau pour chaque nom *X* est créé, où chaque ligne contient :

— un patron syntaxique où *X* apparaît ;
— les expressions selon ce patron où *X* apparaît ;
— les sens possibles de *X* selon ce patron ;
— les probabilités de *X* d'avoir chacun des sens possibles.

Finalement, chaque tableau est connecté à EuroWordNet : chaque sens dans un tableau est connecté au synset qui représente ce sens.

Schwab *et al.* (2007) proposent la création de vecteurs conceptuels pour la représentation de relations syntagmatiques dans WordNet. Voici un exemple du vecteur conceptuel du mot *vie* :

```
V (vie) = (VIE [0.7], NAISSANCE [0.48], ENFANCE [0.46], MORT[0.43],
          VIEILLESSE[0.41], . . .).
```

Les valeurs entre '[]' mesurent l'intensité de la relation entre le concept *vie* et d'autres concepts, et ces vecteurs sont construits en utilisant des algorithmes qui

calculent une mesure de vraisemblance, appliquée sur des dictionnaires, liste de synonymes, etc.

Pour chaque définition dans WordNet, par exemple la définition du mot *ant* (fourmi) : « ant : social insect who lives in an organized colony », des ensembles de mots fonctionnels (verbes, noms, adjectifs et adverbes) liés sémantiquement sont extraits : $x_1 = \{social; insect\}; x_2 = \{organized; colony\}; e_1 = \{live\}$.

Pour chaque mot dans ces ensembles, des vecteurs conceptuels sont calculés. Ensuite, la somme vectorielle est calculée, avec tous les vecteurs conceptuels de la définition du mot (*ant*), en donnant un poids « 1 » aux vecteurs représentant les têtes des ensembles (noms et verbes) et un poids « 0,5 » aux vecteurs représentant les adjoints (adjectifs et adverbes). La somme vectorielle de deux vecteurs, A et B, est donnée par la distance angulaire entre les vecteurs, selon la formule :

$$d = \frac{(A \quad B)}{jAj \quad jBj} \tag{7.1}$$

Où « $A \quad B$ » est calculé sur chaque position des vecteurs ($A \quad B = a_1 \quad b_1 + a_2 \quad b_2 + \quad + a_n \quad b_n$) et $jAj = \sqrt{a_1^2 + a_2^2 + \quad + a_n^2}$

Finalement, le vecteur résultant de la somme vectorielle est attaché au synset dont la définition a été utilisée (dans notre exemple, le synset représentant le mot *ant*).

Nous citons d'autres travaux qui traitent des relations paradigmatiques dans WordNet : Maziarz *et al.* (2012) proposent la création de nouvelles relations paradigmatiques entre adjectifs dans WordNet en polonais et Vincze *et al.* (2008) analysent la possibilité, sans l'implémenter, de la création de relations paradigmatiques absentes de WordNet en utilisant les fonctions lexicales.

7.4 Discussion : représentation de collocations et d'expressions polylexicales

Le tableau 7.4 présente un résumé de différentes méthodes de représentation de collocations et d'expressions polylexicales en général, qui ont été présentés aux sections 7.2 et 7.3.

Tableau 7.4: Résumé des méthodes pour la représentation de collocations et d'expressions polylexicales

Modèle	**Langue**	**FL ?**	**Colloc./EPL/R. syntag**	**Moyen**	**RDF**	**Section**
Villavicencio *et al.* (2004)	anglais	non	EPL (exp idiom, v. à part.)	tableaux	non	7.2
Grégoire (2010)	néerland.	non	EPL	arbre depend.	non	7.2
Multiflex (2010)	anglais	non	EPL	tableaux	non	7.2
Poleng (2010)	anglais	non	EPL	chaîne carac	non	7.2
Lichte *et al.* (2016)	anglais	non	EPL	gram. XMG	non	7.2
Bentivogli et Pianta (2004)	italien	non	r. syntagmatiques	réseau lexical	non	7.3.5
Nica *et al.* (2004)	espagnol	non	r. syntagmatiques	réseau lexical	non	7.3.5
Schwab *et al.* (2007)	anglais	non	r. syntagmatiques	rés lex + vect.	non	7.3.5
Heid et Raab (1989)	anglais	oui	collocation	dictionnaire	non	7.3
Heylen *et al.* (1994)	anglais	oui	collocation	gram. HPSG	non	7.3
Lefrançois et Gandon (2012)	anglais	oui	collocation	dictionnaire	oui	7.3
Lux-Pogodalla et Polguère (2011)	français	oui	collocation	réseau lexical	non	5.7
Lareau *et al.* (2012)	anglais	oui	collocation	gram. XLE	non	7.3

Pour faciliter la comparaison avec notre proposition de représentation de collocations, nous divisons ces travaux en trois groupes :

— Groupe 1 : les quatre premiers travaux montrés au tableau 7.4, qui portent

sur la représentation d'expressions polylexicales ;

— Groupe 2 : les trois travaux suivants, qui traitent de la représentation de relations syntagmatiques dans WordNet ;

— Groupe 3 : les travaux qui traitent la représentation de collocations en utilisant les fonctions lexicales.

Les travaux du groupe 1 traitent la représentation d'expressions polylexicales (EPL) en général. En théorie, leurs méthodes pourraient être utilisées pour représenter des collocations. Une différence par rapport à notre modèle est que ces travaux ne montrent pas comment représenter les différents types de sens prédicatifs (comme *intensification* ou *réalisation*) existant entre les mots qui forment les collocations, tels quels formalisés par les FL. Autre différence est le format de la représentation : des tableaux ou des chaînes de caractères, ce qui nous semble moins intuitif que la représentation des relations de collocations dans un format de graphe, comme dans un réseau lexical.

Dans le groupe 2 se trouvent rassemblés les travaux qui traitent de la représentation de relations syntagmatiques dans WordNet. En dépit de leur similarité avec notre projet — car les relations existant entre les mots dans une collocation sont des relations syntagmatiques —, il y a entre ces travaux et notre proposition des différences significatives.

La méthode proposée par Schwab *et al.* (2007) représente les relations syntagmatiques entre les mots qui sont dans les définitions des *synsets*, au lieu de représenter des relations entre les *synsets*. De plus, tel que noté par les auteurs dans la conclusion de l'article, leur méthode ne permet pas de représenter des relations collocatives entre lexies.

Bentivogli et Pianta (2004) proposent la création d'un nouveau nœud dans Word-

Net pour représenter les collocations. En plus d'être redondantes, car les mots qui sont dans les nouveaux nœuds sont déjà dans WordNet, les relations qui connectent ces nouveaux nœuds avec les *synsets* de WordNet ont toutes le même nom générique, *composé par*, qui ne fournit pas la sémantique des relations entre les lexies dans les collocations.

La méthode proposée par Nica *et al.* (2004) a l'avantage de bien représenter les différents sens qu'un mot peut prendre selon différents types de relations syntagmatiques avec d'autres mots. Cependant, il faut ajouter une structure lourde aux *synsets* (des tables contenant plusieurs colonnes et plusieurs lignes) pour chaque synset dans WordNet.

Le troisième groupe est formé par les travaux qui traitent de la représentation de collocations en utilisant les fonctions lexicales. Heid et Raab (1989) proposent de représenter les collocations dans l'entrée des dictionnaires de la base de la collocation. Heylen *et al.* (1994) et Lareau *et al.* (2012) proposent l'insertion de l'information collocative dans l'entrée de la base de la collocation représentée dans les formats de la grammaire HPSG et de la grammaire XLE, respectivement.

En comparaison avec notre proposition, ces trois méthodes n'utilisent pas une représentation par graphe des collocations, comme dans un réseau lexical. En plus, les propriétés des FL ne sont pas représentées.

Lefrançois et Gandon (2012) proposent la construction d'un dictionnaire employant les formalismes du Web sémantique, en utilisant les FL pour représenter les relations entre les mots. La différence entre leur proposition et la nôtre est que les collocations apparaissent comme des entrées de dictionnaire de la base de la collocation, au lieu de créer une relation directe entre les mots dans un réseau lexical. De plus, les informations sur les FL ne forment pas une ontologie métalexicale indépendante de la représentation des relations, comme dans notre modèle.

Finalement, la proposition de Lux-Pogodalla et Polguère (2011) est la création d'un réseau lexical pour la représentation des relations entre des unités lexicales, en utilisant les FL pour modéliser les différentes relations. En comparaison à notre proposition, ils n'utilisent pas les formalismes du Web sémantique pour modéliser les FL et les collocations, dont l'information est stockée dans les tableaux d'une base de données relationnelle. De plus, ils n'offrent pas une représentation des caractéristiques des FL, ou de comment les FL simples sont combinées pour former des FL complexes.

7.5 Extraction de collocations en utilisant les fonctions lexicales

En général, les travaux sur l'extraction de collocations se basent sur des méthodes statistiques. La majorité suit les idées proposées par les travaux précurseurs suivants :

- Choueka (1988) a proposé une méthode pour l'extraction de collocations basée sur la fréquence des n-grammes (séquence de n mots, où $n > 1$) ;
- Church et Hanks (1990) ont testé l'utilisation de mesures d'association plus sophistiquées, basées sur la théorie de l'information mutuelle ;
- Smadja *et al.* (1996) ont proposé l'extraction de collocations en utilisant un corpus annoté avec des parties du discours et des informations statistiques, comme l'écart type et la moyenne de la distance entre les mots dans une phrase.

Comme nous nous intéressons à la représentation fondée sur les FL, nous nous concentrons à la prochaine section sur les travaux portant sur l'extraction de collocations en utilisant les FL. Ramisch (2012) présente une révision de l'extraction de collocations et d'expressions polylexicales (EPL) en général, sans utiliser les FL.

En ce qui concerne l'extraction d'EPL, nous avons fait des travaux de comparaison de certains outils d'extraction et de différentes mesures d'association (Fonseca *et al.*, 2014; Fonseca et Sadat, 2014).

7.5.1 Algorithmes de classification pour l'extraction de collocations

Moreno *et al.* (2013) utilisent l'algorithme de classification SVM et le dictionnaire des collocations de la langue espagnole DiCE [2], annoté avec des FL, pour classifier les collocations en cinq catégories sémantiques : *intensité*, *phase*, *manifestation*, *cause* et *expérimentation*.

Plusieurs traits sont utilisés pour la création des matrices d'entraînement : les mots dans les phrases où les collocations se trouvent ; la base ; le collocatif ; la paire base-collocatif ; la partie du discours de la base, du collocatif et des mots qui sont deux positions avant et deux positions après la collocation ; le genre ; le nombre ; la personne de la base et celle du collocatif ; la relation syntaxique entre la base et le collocatif, etc.

Kolesnikova (2011) fait l'évaluation de 68 algorithmes de classification supervisée (par ex. : *BayesNet*, *Bagging*, *AdaBoostM1*, etc.), pour la classification de plusieurs collocations de la langue espagnole suivant le patron *verbe-nom* en différentes FL.

Pour la formation des fichiers d'entraînement, des paires *verbe-nom* ont été extraites automatiquement à partir d'un corpus en espagnol. Les paires les plus fréquentes ont été choisies et celles qui représentaient des collocations ont été manuellement annotées avec leurs FL respectives. Les paires qui n'étaient pas des collocations ont été annotées comme *Free word combination* (FWC). Ces collocations sont des instances de huit différentes fonctions lexicales, plus la « fonction » FWC.

---

2. http ://www.dicesp.com/

Ensuite, pour chaque paire *verbe-nom*, les hyperonymes du verbe et du nom ont été extraits de la version espagnole de WordNet. Pour l'extraction des hyperonymes, le sens utilisé pour chaque mot (dans le cas des mots polysémiques) était le sens pris pour le mot dans la collocation. Si aucun hyperonyme n'était trouvé pour un verbe ou pour un nom dans une paire *verbe-nom*, cette paire était exclue de la liste.

Un ensemble d'entraînement a été créé pour chaque FL (il y en a donc neuf au total). Chaque ensemble a été formé par une matrice qui contient une ligne pour chaque paire *verbe-nom*. Dans chaque ligne, chaque position représente un hyperonyme (verbe ou nom) trouvé dans WordNet. S'il s'agit de l'hyperonyme d'un des mots de la paire, il est représenté par « 1 », sinon il est représenté par « 0 ». La dernière position dans la ligne contient la valeur « yes » si la ligne représente une instance de collocation de la FL représentant l'ensemble, et « no » si ce n'est pas le cas.

Par exemple, pour l'ensemble qui représente la fonction $Oper_1$, si la première ligne représente la paire *hacer mención* (mentionner), la première position représente le sens *entidad* (entité) et qu'il est un hyperonyme de *hacer* (faire) ou de *mención* (mention), alors la première position de la première ligne est « 1 ». Et comme *hacer mención* est une collocation modélisée par la FL $Oper_1$, la dernière position de la première ligne est « yes ».

Divers algorithmes de classification ont été appliqués à chaque fichier d'entraînement et de test. La *précision*, le *rappel* et la *F-mesure* ont été calculés pour chaque algorithme et chaque FL, afin de déterminer le meilleur algorithme pour l'identification de chaque FL.

Par exemple, l'algorithme *Bayesian Logistic Regression* a été le plus efficace pour identifier les collocations modélisées par la FL $Oper_1$ et l'algorithme *Prism* a été

le meilleur pour l'identification des collocations modélisées par la FL $Real_1$.

Wanner *et al.* (2006) présentent des techniques qui utilisent des algorithmes de classification pour faire l'attribution des collocations aux FL qui les modélisent. Leurs techniques utilisent l'algorithme de *plus-proche-voisin*, qui peut être utilisé lorsque des collocations *prototypiques* sont définies pour les FL qui les modélisent.

À partir des collocations données en exemples, l'algorithme peut classifier automatiquement les autres collocations. Wanner *et al.* (2006) présentent encore deux autres méthodes de classification, basées sur les réseaux bayésiens naïfs, pour attribuer automatiquement les FL aux collocations.

7.5.2 Outil Colex pour l'extraction de collocations

Orliac (2006) présente l'outil *Colex* pour l'extraction de collocations terminologiques de la langue anglaise, basé sur les FL. Ce système combine les approches symbolique et statistique.

Comme dans le travail de Kolesnikova (2011), les collocations extraites sont du type *verbe-nom* et suivent trois types de relations grammaticales modélisées par les FL (exemples donnés par l'auteure) :

— Sujet + verbe : *The program executes*

— Verbe + premier complément : *[to] close a file*

— Verbe + deuxième complément : *[to] load into memory*

La méthode utilisée pour faire l'extraction est l'appariement entre des règles qui suivent ces types de relation et l'arbre syntaxique obtenu par un analyseur syntaxique appliqué aux phrases d'un corpus en langue anglaise.

Les règles sont formées de deux parties : la condition et l'action. Voici un exemple

de règle (Orliac, 2006, p. 272) :

— Partie condition : if (anyNoun (aux) v_passive (adv) n_obj*)

— Partie action : then {v_passive + anyNoun ;}

Lorsque la règle ci-dessus est appariée à l'arbre syntaxique suivant, qui correspond à la phrase « All files you add to your Web site should be saved with suffix HTML » (Tous les fichiers que vous ajoutez dans votre site Web devront être enregistrés avec le suffixe HTML), on obtient :

— Arbre syntaxique : NP_*file*_pl Aux_*should* V_*saved*_pass NP_suffix_*with*

Et on a les appariements suivants dans la partie condition :

— anyNoun *!* NP_file_pl (*files*)

— v_passive *!* V_saved_pass

— n_obj* *!* NP_suffix_with

Et voici le résultat, dans la partie action :

— Fichier résultat : *!* save *!* file (paire verbe-nom)

Le résultat est déterminé par la partie action dans cet exemple : v_passive (*save*) + anyNoun (*file*).

La précision obtenue en utilisant ces règles appliquées à un corpus textuel en langue anglaise, après la création des arbres syntaxiques, est de 85%.

Cela signifie que 85% de paires extraites sont de vraies paires *verbe-nom*. Cependant, seulement 57% d'entre elles sont des collocations. Par exemple, en utilisant

le nom *file*, on obtient des paires verbe-nom qui sont des collocations : *[to] save a file*, *[to] copy a file*, *[to] open a file* ; et des paires qui ne le sont pas : *[to] be a file* (*it is a file*), *[to] use a file*, *[to] see a file*.

Pour augmenter cette dernière précision, l'auteure utilise un traitement statistique des combinaisons extraites. Elle démontre que la simple fréquence de paires *verbe-nom* n'est pas suffisante pour distinguer entre les paires qui sont et ne sont pas des collocations. Par exemple, la paire (*[to] be*, *file*) est la plus fréquente dans le corpus analysé et elle n'est pas une collocation.

Une alternative possible est l'utilisation de deux mesures d'association : l'information mutuelle et le test du rapport de vraisemblance. Ces dernières mesurent la force d'association entre les mots dans une paire par des formules qui combinent la fréquence de co-occurrences et la fréquence simple des mots.

L'application du test du rapport de vraisemblance a donné de meilleurs résultats : après la classification des paires en utilisant cette mesure, 71% des paires extraites étaient réellement des collocations.

7.5.3 Extraction de collocations en utilisant FrameNet

Alonso Ramos *et al.* (2008) utilisent FrameNet (Fillmore, 1977) pour l'extraction de collocations du type verbe support. FrameNet est basé sur l'idée de cadre (frame). Chaque cadre est une représentation du patron d'une situation abstraite. En voici deux exemples : « Communiquer quelque chose à quelqu'un », « Acheter quelque chose et payer avec un moyen de paiement ». L'unité basique de FrameNet est le *core frame element* (FE), par exemple *Communicateur*, *Évaluateur*, *Expérimentateur*, etc.

Chaque cadre est associé à des unités lexicales qui l'« évoquent ». Par exemple, le cadre *Jugement* est évoqué par des unités lexicales comme *accusation*, *critique*,

etc. Une unité lexicale qui évoque un cadre est appelée *target* (*cible*). Des corpus textuels associés à FrameNet sont annotés avec des noms de FE et des noms de cibles, comme *verbe support*, *contrôleur*, etc.

La méthode d'Alonso Ramos *et al.* (2008) consiste à parcourir automatiquement un corpus annoté avec des cadres et des mots cibles pour repérer les verbes annotés comme verbes supports (*Supp*) et les mots cibles associés à ces verbes supports. Chaque paire (*Supp*, *cible*) est une collocation du type verbe support qui est associée à la FL $Oper_i$.

Ensuite, une heuristique est utilisée pour déterminer l'indice de la fonction $Oper_i$. Par exemple, si le sujet du verbe support est un agent, une personne, etc., l'indice de $Oper_i$ est *1*, donc la fonction est $Oper_1$.

7.6 Autres applications utilisant les fonctions lexicales

Dans cette section nous présentons des applications autres que la représentation et l'extraction des collocations, qui utilisent les FL, comme la traduction automatique, la génération de texte et le paraphrasage.

7.6.1 Traduction automatique

ETAP-3 (Boguslavsky *et al.*, 2004) est un système de traduction automatique basé sur la TST. Il traduit entre une variété de paires de langues (par ex. : l'anglais, le russe, l'arabe, etc.) en utilisant un langage de représentation de sens appelé *Universal Networking Language* (UNL)[3] du côté source ou du coté cible.

Pour chaque langue, il y a un dictionnaire, et chaque entrée de mot dans ce dictionnaire est constituée de deux zones : une générale, comprenant des informations

3. http://www.bibalex.org/en/Project/Details?DocumentID=285

sur la partie du discours, des caractéristiques syntaxiques et sémantiques, etc., et une seconde zone pour chaque langue cible, avec des informations sur la traduction de ce mot dans la langue cible.

Plusieurs entrées contiennent des FL afin d'informer comment elles se connectent à d'autres mots pour former des collocations et comment chaque collocation doit être traduite dans chacune des langues cibles.

Vandaele et Lubin (2005) présentent une approche pour la conceptualisation des collocations métaphoriques en utilisant des FL. Cette approche est destinée une application dans le domaine de la traduction de textes en langage spécialisé, par exemple ceux provenant du domaine biomédical.

7.6.2 Dictionnaire multilingue

Tomokiyo *et al.* (2006) présentent *Papillon*, un dictionnaire multilingue basé sur la TST. Pour chaque langue implémentée (les auteurs citent l'anglais, le français et le japonais), chaque entrée de dictionnaire est représentée avec toutes ses significations. Chaque sens est lié à un concept général représenté dans un dictionnaire en format UNL.

De plus, pour chaque entrée, dans chaque langue, il y a des sections qui informent l'utilisation de ce mot dans différents contextes, comme celui lié aux collocations. Par exemple, il y a les sections : *régime*, *fonctions lexicales*, *formule sémantique*, etc.

7.6.3 Génération de texte

Lareau *et al.* (2011) expliquent comment appliquer les collocations à la génération de textes multilingues, en important des FL dans une grammaire LFG (tel que présenté à la section 7.3.3) et en ajoutant aux dictionnaires multilingues des

informations sur les FL à l'entrée du mot clé des collocations.

Lambrey et Lareau (2015) implémentent, pour les verbes supports, des règles de lexicalisation plus générales que celles implémentées par MARQUIS (Lareau et Wanner, 2007; Wanner *et al.*, 2010). Ces règles sont utilisées pour la génération de collocations et appliquées à la génération automatique de textes multilingues.

7.6.4 Autres applications

Finalement, nous citons le travail d'Apresjan *et al.* (2000), qui fait le résumé des applications inscrites dans le domaine du TALN qui utilisent les FL. Outre l'application à la traduction automatique, la construction de dictionnaires multilingues et la génération de texte, ils citent la désambiguïsation de l'analyse syntaxique, le paraphrasage et l'apprentissage du lexique assisté par ordinateur.

Quelques exemples des règles de paraphrasage qui utilisent les FL ont été donnés à la section 4.8. Voici d'autres exemples de l'utilisation des FL pour le paraphrasage :

— $Oper_1$ + S_0(X) (*They resisted the enemy* – *They o ered resistance to the enemy*) : $Oper_1$(*enemy*) = {*to resist*} ; S_0(*resist*) = {*resistance*}

— $Conv_{12}$(X) (*The group consists of 20 persons* – *Twenty persons comprise this group*) : $Conv_{12}$(*to consist*) = {*to comprise*}

7.7 Conclusion

Dans ce chapitre, nous avons fait la revue des travaux qui traitent de la représentation des expressions polylexicales, des collocations et des relations syntagmatiques en général. Nous avons aussi vu d'autres applications pour les fonctions lexicales, comme l'extraction des collocations à partir de corpus textuels.

La majorité des travaux qui traitent de la représentation des expressions polylexicales est basée sur la représentation en tables ou en chaînes de caractères, sans

l'appui d'une théorie linguistique pouvant modéliser les relations entre les mots.

Les travaux qui traitent de la représentation des relations syntagmatiques sont plutôt basés sur la représentation en vecteurs de mots connectés à WordNet. Normalement, ces représentations ajoutent des informations pertinentes du point de vue combinatoire. Cependant, à ces travaux aussi, il manque un formalisme linguistique pour systématiser les différents types de relations.

Les travaux les plus proches de notre modèle sont ceux qui représentent les collocations en utilisant les fonctions lexicales de la théorie Sens-Texte. La majorité traitent la représentation dans des grammaires, comme HPSG et XLE. À ces formalismes, il manque la flexibilité et la portabilité d'une représentation, sous le format d'une ontologie, par exemple.

Comme nous l'avons vu, les fonctions lexicales peuvent être utilisées pour plusieurs applications, comme l'extraction de collocations, la génération de texte et la traduction automatique. Avoir une représentation de collocations qui les utilise constitue un avantage en termes des possibles applications et de systématisation sémantique et syntaxique.

Au chapitre suivant, nous présentons *lexfom*, notre modèle pour la représentation des fonctions lexicales et des collocations. Nous montrons, à l'aide d'exemples, comment une collocation peut être représentée dans un réseau lexical avec ce modèle.

CHAPITRE VIII

MODÈLE LEXFOM

8.1 Introduction

Nous présentons dans ce chapitre le modèle *lexfom* (Lexical functions ontology model) (Fonseca *et al.*, 2016a,b), une ontologie métalinguistique basée sur les formalismes du Web sémantique (Antoniou et van Harmelen, 2004), pour la représentation des fonctions lexicales (simples standard et complexes) et des relations lexicales (paradigmatiques et syntagmatiques).

Nous avons extrait du Réseau Lexical du Français (RLF) (Lux-Pogodalla et Polguère, 2011) les informations concernant les FL représentées dans notre modèle. Nous avons aussi extrait du RLF les relations sémantiques qui forment de collocations, modélisées par des FL, pour la langue française.

Lexfom est divisé en quatre modules :

— La section 8.2 présente le module *lfrep* (*lexical function representation*), qui est l'implémentation des propriétés des FL ;

— La section 8.3 présente le module *l am* (*lexical function family*), créé pour l'organisation des FL en classes syntaxiques/fonctionnelles similaires ;

— La section 8.4 présente le module *lfsem* (*lexical function semantic perspective*), pour l'encodage des perspectives sémantiques des FL, tel que présenté

dans la section 4.7 ;

— La section 8.5 présente le module *lfrel* (*lexical function relations*), pour la représentation des relations entre unités lexicales.

Cette division modulaire suit le patron utilisé par *lemon*. Elle sépare mieux les différents types de connaissances représentées et facilite la maintenance et l'amélioration du modèle.

Dans les sections suivantes, nous présentons les quatre modules. Dans la section 8.6 nous montrons comment *lexfom* peut être utilisé dans la représentation de collocations dans un réseau lexical.

Dans la section 8.7 nous comparons notre modèle avec les travaux présentés dans le chapitre 7 pour la représentation des collocations.

8.2 Module de représentation de fonctions lexicales

Dans cette section, nous présentons le module *lfrep* (*lexical function representation*), qui représente les propriétés des FL. La figure 8.1 montre les classes qui composent ce module et les propriétés qui connectent ces classes.

Une FL est représentée par la classe *lexical_function*. Elle se connecte à la classe *lexicalFunctionFamily* pour indiquer à quelle famille la FL appartient, tel qu'expliqué dans la section 8.3. Elle se connecte à la classe *semanticPersp* pour indiquer le sens abstrait d'une FL, tel que vu dans la section 8.4. Une FL peut avoir une ou plusieurs perspectives sémantiques.

Dans toutes les figures de ce chapitre, représentant le modèle *lexfom*, chaque rectangle représente une classe. Les noms en dessous de la ligne qui divise le rectangle sont des instances de la classe respective, à l'exception des classes *compositionalType* et *constituentLF*, où les noms en dessous de la ligne représentent des

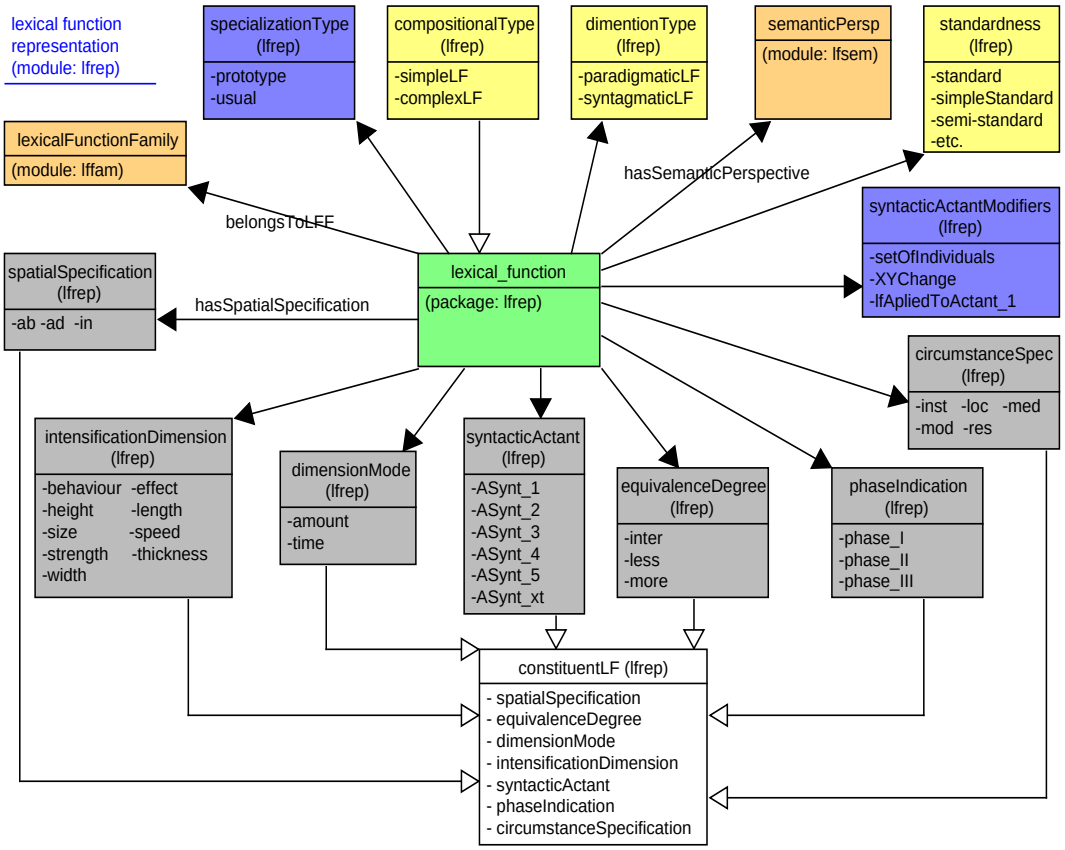

Figure 8.1: Lexical function representation module

sous-classes.

Les classes en jaune, sur le dessus de la figure, représentent des propriétés communes aux FL en général. Les FL peuvent être sous-catégorisées par ces propriétés. Par exemple, nous pouvons séparer les FL en simples et complexes, paradigmatiques et syntagmatiques, standards, semi-standards et non-standards, etc.

Ces trois classes sont l'implémentation des regroupements des FL présentés dans la section 4.4. Les trois classes sont :

— *dimentionType*

- *paradigmaticLF*
- *syntagmaticLF*

— *standardness*

- *standard*
- *simple_standard*
- *semi-standard*
- *locally_standard*
- *locally_semi-standard*
- *proposed_standard*
- *proposed_simple_standard*

— *compositionalType*

- *simpleLF*
- *complexLF*

La flèche blanche connectant la classe *compositionalType* à la classe *lexical_function* indique que ce dernière classe est une sous-classe de la première.

La classe *lexical_function* est connectée aux classes *dimentionType* et *standardness* par des propriétés, indiquées par des flèches noires.

Les classes en bleu représentent des types spéciaux de FL. Elles sont l'implémentation de FL présentées dans la section 4.5. Il y a deux classes pour les représenter :

— *specializationType*

- *prototype*
- *usual*

— *syntacticActantModifiers*

- *setOfIndividuals*
- *XYChange*
- *lfAppliedToActant*

Les classes au bas de la figure 8.1 sont des sous-classes de la classe *constituentLF* et représentent des propriétés spécifiques de certaines FL, tel que présenté dans la section 4.6. Par exemple, la classe *syntacticActant* définit les actants syntaxiques possibles de certaines FL (par ex., les FL $Oper_i$, $Real_i$, $Func_i$, etc).

Il y a sept de ces classes :

— *circumstanceSpec* : *inst*, *loc*, *med*, *mod*, *res*

— *dimensionMode* : *amount*, *time*

— *equivalenceDegree* : *inter*, *less*, *more*

— *intensificationDimension* : *behaviour*, *e ect*, *height*, *length*, *size*, *speed*, *strength*, *thickness*, *width*

— *phaseIndication* : *phase_I*, *phase_II*, *phase_III*

— *spatialSpecification* : *ab*, *ad*, *in*

— *syntacticActant* : *ASynt_1*, *ASynt_2*, *ASynt_3*, *ASynt_4*, *ASynt_5*, *ASynt_xt* (actant externe)

La figure 8.2 montre comment le module *lfrep*, combiné avec les modules *I am* et *lfsem*, est utilisé pour la représentation en RDF de la FL $Oper_1$.

Dans ce code, nous observons l'utilisation de deux autres modules *I am* et *lfsem*, pour indiquer la famille de la FL ($Oper_1$) et sa perspective sémantique (*verbe support*), respectivement. C'est aussi là qu'est encodée l'information que cette FL possède l'actant syntaxique *1* et que cette FL est syntagmatique.

```
LF-Oper1 rdf:type lfrep:simpleLF, owl:NamedIndividual;
   lfrep:belongsToLFFamily lffam:LFF-synt-suppV-Oper1;
   lfrep:hasSyntActant lfrep:lfrep-const-sa-ASynt_1;
   lfrep:dimension lfrep:lfrep-type-syntagmaticLF;
   lfrep:semanticPerspective lfsem:pSem-sv-supportVerb.
```

Figure 8.2: Code RDF représentant la FL *Oper*$_1$

Le fait que la FL *Oper*$_1$ appartient à la famille *Oper1* n'est pas une redondance. Cette famille contient aussi des FL complexes, formées par la combinaison de la FL *Oper*$_1$ et d'autres FL, tel qu'expliqué dans la section 8.3.

8.2.1 Représentation des fonctions lexicales complexes

La représentation des FL complexes fait aussi partie du module *lfrep*. Pour les représenter, il y a une classe de plus, *component*, et trois propriétés : *constituent*, *correspondsTo* et *compositionType*.

Chaque FL complexe (sous-section 4.4.3) est représentée par de composants. Chaque composant correspond à une des FL simples ou complexes qui forment la FL complexe. Ils sont également connectés par une propriété qui indique le type de composition.

La figure 8.3 montre comment les FL complexes sont représentées.

La figure 8.4 montre la représentation de la FL complexe *CausIncepOper*$_1$ comme exemple. Elle est représentée par l'objet *LF_CausIncepOper1*, qui est connecté par des propriétés *constituent* à deux composants. Dans ce figure, chaque rectangle représente une de instance des classes montrées par la figure 8.3.

Le premier composant correspond à la FL simple *Caus* et le second à la FL complexe *IncepOper*$_1$. Cette dernière est connectée à deux autres composants, qui correspondent à la FL *Incep* et à la FL *Oper*$_1$.

Figure 8.3: Représentation des fonctions lexicales complexes

Il faut souligner que les représentations des FCaus, Incep, $Oper_1$ et $IncepOper_1$ sont indépendantes de la représentation de la FCausIncepOper$_1$. Pendant la création de la représentation de cette dernière FL, les deux composants nécessaires sont créés. Après la création des composants, ils sont connectés par des propriétés correspondsTo aux représentations d'autres FL qui ont été déjà créées auparavant.

Finalement, notons la propriété compositionType connectant les différents composants. Cette propriété modélise les types de composition des FL complexes, expliqués dans la sous-section 4.4.3.

Figure 8.4: Représentation de la fonction lexicale $CausIncepOper_1$

8.3 Module de familles de fonctions lexicales

Dans cette section, nous présentons le module lfam (Lexical function family), créé pour le regroupement des FL par des similarités syntaxiques. L'origine du concept de famille de FL est le Réseau Lexical du Français (RLF) (Section 5.7). La figure 8.5 montre le modèle de ce module.

Les FL sont divisées en deux grands groupes : les FL syntagmatiques (celles qui représentent des collocations) et les FL paradigmatiques. Chacun de ces groupes est divisé en sous-classes.

La classe paradigmaticLF est divisée en neuf classes :

Figure 8.5: Module lexical function family

actantialAdjectives : A_1, A_2, A_3, A_4

actantialAdverbs : Adv_1, Adv_2, Adv_3

actantialNouns : S_1, S_2, S_3, S_4

circumstantialNouns : S_{instr} , S_{loc}, S_{med}, S_{mod}, S_{res}

potentialActAdj : $Able_1$, $Able_2$, $Able_3$, $Able_4$

qualifyingActAdj : $Qual_1$, $Qual_2$

resultative : $Result_1$, $Result_2$, $Result_3$

setPart : Holo, Hyper, Hypo, Mero

syntacticConversion: A_0, Adv_0, S_0, V_0

La classe syntagmaticLF est divisée en quatre classes :

supportVerbs : Oper, Func, Labor

realizationVerbs : Real, Fact, Labreal

phasalVerbs : Incep, Cont, Fin

causationVerbs : Caus, Perm, Liqu

Il y a aussi des FL qui n'appartiennent pas à une de ces treize classes. Elles sont placées directement dans la classe syntagmaticLF ou dans la classe paradigmaticLF. Par exemple les FL Magn, Bon, AntiMagn (syntagmatiques) et les FL Anti, Conv, Syn (paradigmatiques).

Chaque famille contient une ou plusieurs FL simples et un grand nombre de FL complexes. Par exemple, la famille Oper$_1$ contient, parmi d'autres, les FL : Oper$_1$, CausOper$_1$, FinOper$_1$, CausIncepOper$_1$, NonOper$_1$, S$_1$Oper$_1$, etc.

Cette classification est utile, par exemple, pour les requêtes. Il est possible de chercher des collocations formées par des verbes de phase, des collocations adjectivales ou adverbiales seulement, des collocations qui dénotent le sens d'intensification (la famille Magn), etc.

Finalement, la requête pour des collocations qui dénotent les sens de qualification subjective positive (Bon) et qualification subjective négative (AntiBon) sera très utile dans des applications liées au TALN, comme l'analyse des sentiments.

8.4 Module de perspective sémantique de fonctions lexicales

Nous présentons dans cette section le module lfsem (Lexical function semantic perspective). Il a été conçu pour connecter un sens abstrait à une FL. Son modèle est représenté par la figure 8.6.

Le moduleLfsem est une implémentation de la perspective sémantique développée par Jousse (2010), présentée dans la section 4.7.

Figure 8.6: Module lexical function semantic perspective

Ce module représente une classi cation des possibles sens d'une FL. Par exemple, la classeactionEvent contient les sens :attempt, creation, manifestation, typicalOperation, etc.

Un autre exemple : les FLMagn et Bon, qui dénotent les sens abstraits d'intensi - cation et de quali cation subjective, respectivement, appartiennent à la classequali - li cation . Cependant, la FL Magn est connectée à l'instance:intensity (intensité) de la classequali cation , tandis que la FL Bon est connectée à l'instance:judg-

ment (jugement). Les instances de la classe lexical_function sont connéctées aux instances de la classe semanticPersp par des propriétés hasSemanticPerspective

Notre modèle implémente 12 classes que représentent di érentes perspectives sémantiques. Les classes et les membres de ces classes sont les suivantes :

actionEvent : attempt, creation, decreaseDegradation, disparationExistencialCease, imminence, increaseImprovement, manifestation, nonOperationRefusal, utilizationTypicalOperation

causativity : causativity

elementSet: family, hierarchies, leaderTeam, measure, meronymyHolonymy regularElementSet

equivalence: conversionEquivalence, similarLexies, syntacticConversion

form : instrument, realizationForm, utilizationForm

location : spatialTemporal, typicalPlace

opposition : conversionOpposite, genderEquivalent, semanticOpposite

participants : externalActant, keywordActant

phaseAspect: continuation, duration, preparation, reiteration, resultPhase, start, termination

quali cation : intensity, judgement, physicalAppearance

semanticallyEmptyVerb: semanticallyEmptyVerb

supportVerb : supportVerb

Ce module représente une façon de regrouper di érentes FL dont les sens abstraits sont similaires et, en conséquence, regrouper les collocations et les relations sémantiques qu'elles représentent.

8.5 Module de représentation de relations lexicales

Nous présentons dans cette section le module lfrel (Lexical function relation), représenté dans la figure 8.7. L'objectif de ce module est de représenter la connexion entre deux unités lexicales, représentées par des objets lemon lexicalSense, ayant entre eux une relation paradigmatique ou syntagmatique.

La classe lfSenseRelation représente une relation syntagmatique ou une relation paradigmatique. Cette classe est connectée à la classe lexicalFunction, qui appartient au module lfrep, pour indiquer la FL qui représente la relation. Chaque lexicalSense est connectée à une classe lemon lexicalEntry. Différentes informations syntaxiques, morphologiques et lexicales peuvent être connectées à un objet lexicalEntry, comme expliqué dans la section 6.5.2.

La figure 8.8 montre le code RDF qui représente la collocation vent puissant. Le sens qui connecte les deux unités lexicales pour former la collocation est modélisé par la FL Magn (intensification) : Magn(vent) = {puissant}.

Les indices I.1 et II.1, connectés aux sens vent et puissant, indiquent des sens spécifiques (lexèmes) des vocables vent et puissant, tel que vu dans la section 1.9 et dans la section 6.5.2.

La classe senseRelationDirection est utilisée pour indiquer la direction (mot-clé - valeur ou valeur - mot-clé) dans une relation syntagmatique. Par exemple, dans la collocation peur bleue, peur est le mot-clé et bleue est la valeur (collocatif). La direction de cette relation est keywordValue (mot-clé - valeur).

Par contre, dans la collocation faire attention, attention est le mot-clé et faire est la valeur (collocatif) et la direction de cette relation est valueKeyword (valeur - mot-clé).

Figure 8.7: Module lexical function relation

```
:lfsr_23734 a lfrel:SyntagmaticLFSenseRelation;
        lfrel:hasLexicalFunction lfrep:LF-Magn;
        lfrel:hasLFKeyword ontolex:vent_sense_I.1;
        lfrel:hasLFValue ontolex:puissant_sense_II.1;
        lfrel:hasFusedElement "false"^^xsd:boolean.
```

Figure 8.8: Code RDF représentant la relation syntagmatique entre les unités lexicalesvent_I.1 et puissant_II.1 qui forme la collocationvent puissant

La classegovernmentPattern indique le régime d'une collocation. Le régime représente la position des prépositions, articles et éléments externes en relation au mot-clé et au collocatif.

Par exemple, la collocationporter un vêtement est modélisée par la FLReal$_1$

comme suit : Real(vêtement) = { porter DET };

Dans cet exemple, le régime est DET }, où DET représente un déterminant (article dé ni ou indé ni) et représente le mot-clé de la collocation.

L'information sur la direction de la relation lexicale et sur le régime est importante pour les applications de génération de textes et de traduction automatique, notamment.

8.6 Exemple complet de la représentation d'une collocation

Dans cette section, nous montrons une application de notre modèle, la conversion du Réseau Lexical du Français (RLF) (Lux-Pogodalla et Polguère, 2011) dans un format d'ontologie.

Le RLF est représenté dans un format de base de données et son contenu peut être extrait en utilisant le langage de requête SQL (Structured Query Language). Nous avons extrait du RLF l'information concernant les FL, les vocables, les unités lexicales et les relations syntagmatiques entre les unités lexicales.

Nous représentons chaque unité lexicale dans le RLF comme un objet lexfom. Chaque FL et chaque relation syntagmatique est représentée comme des objets lexfom.

Comme exemple, nous montrons la représentation de la collocation porter un vêtement. Dans la section 5.7 nous montrons comment cette collocation est représentée dans le RLF. La gure 8.9 montre la représentation de cette collocation en utilisant le module lfrel.

Nous montrons aussi :

- la représentation de chaque unité lexicale (porter et vêtement) comme un

Figure 8.9: Représentation de la collocation *porter un vêtement* en utilisant le module lfrel

objet lemon LexicalEntry et comme un objet lemon LexicalSense ;

la représentation de la FL qui connecte les deux unités lexicales pour former la collocation : $Real_1$(vêtement) = {porter DET } ;

la représentation de la relation lexicale.

En utilisant le modèle lexfom, la représentation de cette collocation est comme suit :

Premièrement, la gure 8.10 montre le code RDF représentant le vocable vêtement. Nous avons extrait du RLF cinq sens (lexèmes) de ce vocable

qui participent en collocations. Chaque sens est représenté comme un objet lemon LexicalSense ;

Deuxièmement, la figure 8.11 montre le code RDF représentant le vocable porter. Nous avons extrait du RLF deux sens (lexèmes) de ce vocable qui participent en collocations. Chaque sens est représenté comme un objet lemon LexicalSense ;

Ensuite, la figure 8.12 montre le code RDF représentant la FL $Real_1$;

Finalement, la figure 8.13 montre le code RDF représentant la relation syntagmatique entre porter_IV et vêtement_I.2 pour représenter la collocation porter un vêtement.

```
:lex_vêtement a ontolex:LexicalEntry, ontolex:Word;
        ontolex:canonicalForm :form_vêtement;
  ontolex:sense :vêtement_sense_I.2;
  ontolex:sense :vêtement_sense_I.1;
  ontolex:sense :vêtement_sense_II;
  ontolex:sense :vêtement_sense_III.1;
  ontolex:sense :vêtement_sense_III.2;
        rdfs:label "vêtement"@fr .

:form_vêtement a ontolex:Form;
        ontolex:writtenRep "vêtement"@fr .

vêtement_sense_I.2 a ontolex:LexicalSense .
vêtement_sense_I.1 a ontolex:LexicalSense .
vêtement_sense_II a ontolex:LexicalSense .
vêtement_sense_III.1 a ontolex:LexicalSense .
vêtement_sense_III.2 a ontolex:LexicalSense .
```

Figure 8.10: Code RDF représentant le vocable vêtement avec ses cinq sens trouvés dans le RLF

Nous observons que la représentation de la relation syntagmatique n'inclut pas d'informations syntaxiques ou sémantiques, à l'exception du régime, qui est représenté par une chaîne de caractères. Pourtant, toutes les informations syntaxiques et sémantiques sont déjà, ou peuvent être, encodées dans la représentation des unités lexicales comme des objets LexicalEntry et dans la représentation de la FL.

Cette approche présente de nombreux avantages. Par exemple, il n'est pas nécessaire de répéter l'information syntaxique et sémantique donnée par la FL. La

```
:lex_porter a ontolex:LexicalEntry, ontolex:Word;
      ontolex:canonicalForm :form_porter;
            ontolex:sense :porter_sense_I.1;
            ontolex:sense :porter_sense_IV;
      rdfs:label "porter"@fr .

:form_porter a ontolex:Form;
       ontolex:writtenRep "porter"@fr .

porter_sense_I a ontolex:LexicalSense .
porter_sense_II a ontolex:LexicalSense .
```

Figure 8.11: Code RDF représentant le vocable *porter* avec ses deux sens trouvés dans le RLF

```
LF-Real1 rdf:type lfrep:simpleLF, owl:NamedIndividual ;
    lfrep:belongsToLFFamily lffam:LFF-synt-realV-Real1;
    lfrep:hasSyntActant lfrep:lfrep-const-sa-ASynt_1;
    lfrep:dimension lfrep:lfrep-type-syntagmaticLF;
    lfrep:semanticPerspective
      lfsem:pSem-ae-utilizationTypicalOperation.
```

Figure 8.12: Code RDF représentant la fonction lexicale $Real_1$

```
:lfsr_11420 a lfrel:SyntagmaticLFSenseRelation;
       lfrel:hasLexicalFunction lfrep:LF-Real1;
       lfrel:hasLFKeyword ontolex:vêtement_sense_I.2;
       lfrel:hasLFValue ontolex:porter_sense_IV;
       lfrel:hasGovPattern lfgpat:"DET ~s"^^xsd:string;
       lfrel:relationDirection lfrel:valueKeyword;
       lfrel:hasFusedElement "false"^^xsd:boolean.
```

Figure 8.13: Code RDF représentant la collocation *porter un vêtement*

même FL (ici, $Real_1$) peut être attachée à différentes relations lexicales/sémantiques qui forment différentes collocations modélisées par cette FL.

En outre, il n'est pas nécessaire de répéter la représentation de chaque vocable et chaque sens. Ils ne sont représentés qu'une seule fois. Pour chaque collocation, nous avons seulement besoin de créer un nouvel objet *SyntagmaticLFSenseRelation*, le connecter à la représentation de la FL que modélise la relation, le connecter aux deux objets *lexicalSense* représentant chacune des unités lexicales dans la collocation et indiquer la direction de la relation (valeur - mot-clé ou mot-clé -

valeur).

Puisque chaque unité lexicale peut participer à de nombreuses collocations, avec di érentes FL, on peut visualiser un réseau lexical où les unités lexicales sont les n÷uds et les relations syntagmatiques, représentant les collocations, sont les arcs.

Le RLF encode environ 600 FL standard (environ 100 FL simples et 500 FL complexes) et nous les avons toutes encodées en utilisant lexfom. Parmi ces 600 FL, il y a environ 330 types di érents de FL syntagmatiques (simples et complexes) : celles qui représentent le type de relation qui existe entre les unités lexicales dans une collocation.

Ces 330 FL syntagmatiques sont utilisées pour décrire environ 8 000 relations syntagmatiques de la langue française, encodées en utilisant lexfom comme des objets SyntagmaticLFSenseRelation.

8.7 Comparaison entre notre modèle et d'autres formalismes pour la représentation des collocations et des EPL

Nous présentons dans cette section les di érences principales entre notre modèle et les travaux présentés dans la section 7.3 pour la représentation de collocations en utilisant les FL et les travaux présentés dans la section 7.2 pour la représentation des EPL en utilisant d'autres formalismes.

Premièrement, la comparaison entre les travaux qui utilisent les FL pour la représentation des collocations. Les principales di érences sont les suivantes :

- Ces travaux indiquent, dans une entrée de dictionnaire d'un mot m_1 (ou dans la représentation MAV d'une collocation), quels sont les mots qui forment des collocations avec m_1. Ils indiquent aussi les FL qui relient ces mots pour former une collocation. Dans notre modèle, puisqu'il est destiné à encoder

des réseaux lexicaux et non des dictionnaires, les informations sur les collocations ne sont pas indiquées dans des entrées des mots. L'information sur la collocation, modélisée par la classe lexicalSenseRelation, est une entité en soi-même. C'est dans la représentation de la relation qu'il est indiqué quelles unités lexicales forment la collocation. L'avantage est la possibilité de connecter d'autres informations pertinentes à une collocation directement à la collocation et non à son mot-clé.

Dans notre modèle, il y a une représentation des FL, inexistante dans d'autres modèles, qui est organisée par des classes syntaxiques et sémantiques. Ces représentations des FL sont connectées aux représentations des collocations.

Notre modèle s'intègre à lemon, un formalisme reconnu par l'organisation W3C pour la représentation des informations lexicales sur le Web sémantique. Notre modèle profite des informations linguistiques implémentées par lemon ou par d'autres formalismes utilisés par lemon, comme l'ontologie ISOCat.

Puisque lexfom est basé sur les formalismes du Web sémantique (RDF/OWL), un réseau lexical qui utilise notre modèle pour représenter des collocations peut aussi être connecté à d'autres ontologies de domaines différents. De plus, comme le langage OWL est basé sur une logique formelle, il est possible pour des systèmes qui utilisent notre modèle de faire des inférences sur des collocations.

Lorsqu'on compare notre modèle avec la manière dont le RLF représente les collocations, la principale différence est que notre modèle permet de représenter les connexions entre des unités lexicales comme une ontologie, plutôt que de représenter toutes ces informations dans des tableaux de base de données. Cela facilite l'intégration d'un réseau lexical basé sur notre modèle avec d'autres ressources lexicales sur le Web sémantique.

Encore en comparaison avec le RLF, notre modèle implémente une nouvelle forme de classi cation sémantique des FL.

Lorsqu'on compare notre modèle avec d'autres formalismes utilisés pour représenter les EPL qui ne sont pas basés sur les FL, les di érences principales sont les suivantes :

Notre modèle est basé sur les FL, qui est un outil linguistique qui représente les collocations d'une manière plus complète et systématique, syntaxiquement et sémantiquement.

Ces formalismes sont souvent basés sur des tableaux et des séquences de caractères pour représenter les informations syntaxiques liées aux EPL, alors que notre modèle est dans un format d'ontologie. Les informations syntaxiques sont représentées indépendamment de la représentation de la collocation.

L'information syntaxique est mélangée avec l'information sur les EPL. Dans lexfom, l'information syntaxique est encodée dans la représentation des FL connectées aux relations représentant des collocations.

Les types d'EPL non représentés par notre modèle peuvent être traités par lemon, auquel notre modèle est connecté, incluant la représentation du comportement syntaxique des lexemes formant des EPL.

Finalement, soulignons l'originalité de notre travail :

Il est le premier à représenter des propriétés des FL, plutôt que de simplement utiliser le nom des FL dans des relations lexicales. Cette représentation, en format d'ontologie, peut être utilisée dans la représentation de relations syntagmatiques et paradigmatiques dans des réseaux lexicaux, même ceux qui ne sont pas basés sur les FL.

Il est le premier à combiner la représentation des FL à une ontologie métalinguistique (lemon) qui est une recommandation du W3C pour la représentation lexicale sur le Web. Cela permet la connexion de l'information concernant les FL aux informations représentées par des ontologies de n'importe quels domaines.

Il est le premier à représenter, sous le format d'une ontologie, une classification sémantique des FL.

Nous avons créé, pour la première fois, une représentation sous forme d'ontologie des relations syntagmatiques et paradigmatiques présentes dans un réseau lexical basé sur les FL : le RLF.

8.8 Conclusion

Nous avons présenté dans ce chapitre le Lexical functions ontology model (lexfom), une ontologie métalinguistique pour la représentation des fonctions lexicales de la théorie Sens-Texte et la représentation des relations syntagmatiques et paradigmatiques, y compris les collocations. Lexfom se combine à lemom, un patron du W3C pour la représentation des informations linguistiques sur le Web sémantique.

Nous avons vu comment lexfom est divisée en quatre modules : lfrep, lfam, lfsem et lfrel, et comment ces modules sont utilisés pour représenter les informations sur les fonctions lexicales, leurs classifications syntaxiques et sémantiques et les relations lexicales paradigmatiques et syntagmatiques.

Cette ontologie a été appliquée au Réseau Lexical du Français (RLF) pour le transformer dans un format compatible avec le Web sémantique et passible d'être connecté à d'autres réseaux lexicaux en format d'ontologie et des ontologies d'autres domaines.

CHAPITRE IX

IDENTIFICATION DE COLLOCATIONS EN UTILISANT LE RLF SOUS LA FORME D'ONTOLOGIE ET L'ANALYSE DE DÉPENDANCE

9.1 Introduction

Dans ce chapitre, nous présentons une application de notre implémentation du RLF sous la forme d'ontologie (Fonseca et al., 2017). Pour qu'elle soit alignée avec le reste de la thèse, l'application choisie est l'identification de collocations à partir d'un corpus en français.

Nous proposons une méthode pour identifier des collocations basées sur l'analyse syntaxique de dépendance et nous extrayons des candidats appartenant à 14 différentes dépendances (par exemple sujet-objet, verbe auxiliaire causal, modificateurs adjectivaux, etc.).

Pour filtrer les candidats, au lieu d'utiliser des mesures d'association, nous utilisons le RLF sous la forme d'une ontologie. Puis, nous classons les collocations identifiées selon les FL qui les modélisent et selon la perspective sémantique de chaque FL.

Notre principal objectif est d'identifier précisément les collocations plutôt que d'avoir un classement des probables collocations, comme c'est le cas dans les méthodes basées sur des mesures d'associations. De plus, nous identifierons les principaux groupes sémantiques auxquels appartiennent les collocations. Finalement,

nous croyons qu'il est important d'identi er les dépendances syntaxiques qui sont les plus susceptibles de faire partie d'une collocation, puisque la TST et les FL sont basées sur la syntaxe de dépendance.

Un autre point important est que la plupart des travaux liés à la sémantique distributionnelle (Mcdonald et Ramscar, 2001) traitent des relations paradigmatiques, telles que la synonymie et l'hyperonymie. Les relations syntagmatiques sont presque toujours complètement ignorées. (Sahlgren, 2006). Par conséquent, nous croyons qu'une étude montrant la distribution sémantique des relations syntagmatiques, encodées par les collocations et les FL, peut améliorer les études liées à la sémantique distributionnelle.

Il faut noter que le RLF est encore en développement et ne représente qu'un échantillon de vocables, lexèmes et relations lexicales de la langue française. Notre encodage du RLF contient 14 680 vocables et 27 143 lexèmes. IL s'agit d'un facteur limitant à la portée du travail présenté dans ce chapitre.

Finalement, nous soulignons que les vocables extraits du corpus n'étaient pas désambiguïsés.

9.2 Méthodologie

Dans cette section, nous présentons notre méthode pour l'identi cation de collocations. Premièrement, les candidats sont extraits à partir d'un corpus auquel une analyse de dépendance syntaxique a été appliquée. Deuxièmement, nous utilisons notre codi cation du RLF en format d'ontologie pour identi er les vraies collocations parmi les candidats. Troisièmement, nous utilisons notre ontologie pour classer les collocations identi ées selon les di érentes perspectives sémantiques des FL qui les modélisent.

La gure 9.1 présente notre chaîne de traitement pour l'extraction de collocations

à partir d'un corpus de textes.

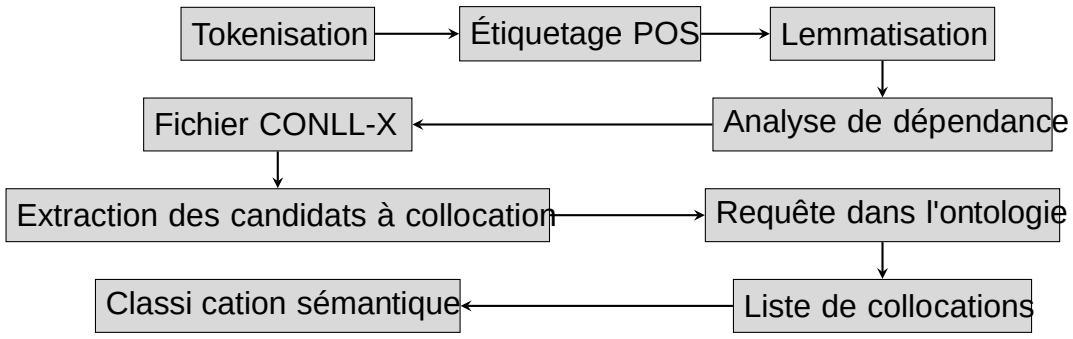

Figure 9.1: La chaîne de traitement pour l'identi cation et la classi cation des collocations

Nous divisons l'explication de la méthodologie en trois parties :

l'utilisation de l'analyse de dépendance pour faire l'extraction des candidats à collocation : dans la gure 9.1, cela correspond au prétraitement (tokenisation, étiquetage POS et lemmatisation), à l'analyse de dépendance, à la génération du chier en format CONLL-X et à l'extraction des candidats à collocation ;

l'utilisation de notre ontologie pour ltrer les candidats et identi er les collocations : dans la gure 9.1, cela correspond à la requête dans l'ontologie et à la génération de la liste de collocations ;

nalement, la classi cation sémantique des collocations.

9.2.1 Utilisation de l'analyse de dépendance pour extraire les candidats

Avant l'extraction des candidats, nous e ectuons un prétraitement dans le corpus : tokenisation, étiquetage en parties du discours et lemmatisation. Après le prétraitement, une analyse de dépendance est appliquée et un chier en format CONLL-X (Buchholz et Marsi, 2006) est généré.

CONLL-X a été créé pour la représentation des dépendances syntaxiques dans une phrase. Dans ce format, chaque token d'une phrase est représenté par une ligne dans une table contenant 10 colonnes. Les informations pertinentes à cette étude sont les suivantes :

- identi cateur (position) de t dans la phrase ;
- forme de surface ;
- forme canonique (lemme) ;
- partie du discours (POS) ;
- identi cateur (ID) du token qui est le gouverneur d'une relation de dépendance syntaxique avec t dans la phrase (zéro si t est la racine de l'arbre de dépendance syntaxique dans la phrase) ;
- type de relation de dépendance entre t et le gouverneur de la relation de dépendance.

Nous faisons l'extraction des candidats à collocation à partir du chier CONLL-X. Sont extraites seulement les paires de mots ayant entre eux une des relations de dépendance suivantes (Marie Candito et Seddah, 2011; Urieli, 2013) :

- a_obj : argument introduit par à - à fond ;
- arg : argument (utilisé dans certaines expressions liées par une préposition) - comme tout;
- ats : prédicatif adjectif ou nominal sur le sujet (attribut du sujet) - être victime ;
- aux_caus : verbe auxiliaire causatif - faire dégager;
- aux_tps : verbe auxiliaire de temps - avoir vu ;

coord : relation portée par un coordonnant, ayant comme gouverneur le premier coordonné. Par exemple, dans le syntagme le garçon et la fille, il y a une relation coord entre le coordonnant et et le coordonné immédiatement précédent garçon ;

de_obj : argument introduit par de - souvient de ;

dep : dépendance non spécifiée - très grave ;

dep_coord : relation portée par le second coordonné, ayant comme gouverneur le premier coordonnant. Dans la phrase ci-dessus, donnée comme exemple pour la dépendance coord, il y a une relation dep_coord entre le coordonné fille et le coordonnant et qui le précède immédiatement ;

mod : modificateurs (nominaux, adjectivaux et adverbiaux repérés structuralement, autres que les relatives) - politique véritable ;

mod_rel : relie un pronom relatif antérieur au verbe régissant une phrase - série qui plaît, mod_rel(série, plaît) ;

obj : objet d'un verbe - traiter les maladies ;

p_obj : argument introduit par une préposition autre que à et de - sur la table ;

suj : sujet d'un verbe - le bateau naviguait ;

Ces dépendances ne sont pas les mêmes que les dépendances utilisées dans Universal Dependencies (UD) [1], qui est un modèle fréquemment utilisé dans les analyses de dépendance syntaxique. Le but de ce projet est d'avoir un ensemble d'étiquettes de dépendances qui sont identiques pour toutes les langues et faciliter le partage et la comparaison des informations.

1. http ://universaldependencies.org/

Notre méthode est basée sur un réseau lexical basé sur des FL, qui, pour le moment, n'existe que pour la langue française. Pour l'analyse de dépendance, le modèle de langue française que nous avons à disposition[2] a été développé avant la création de l'UD. Cela explique pourquoi les balises de dépendances dans notre fichier CONLL-X ne sont pas les mêmes que les balises de l'UD.

9.2.2 Utilisation de notre ontologie pour identifier les collocations

Pour filtrer les candidats et identifier les vraies collocations, nous utilisons le RLF en format d'ontologie. Pour accéder à l'ontologie, nous avons développé une API Java (Fonseca et al., 2018). Les fonctions de l'API pertinentes sont :

searchSyntagRelation(gov, dep) → FL (base, collocatif) : la paire (gouverneur, dépendante) (ou n'importe quelle paire des mots) est cherchée dans la partie de l'ontologie représentant les relations syntagmatiques (RS). S'il y a au moins une RS entre elles, le résultat de la requête retourne la FL qui les relie et l'information sur quel est le mot est à la base et quel est le collocatif dans cette relation.
Example : searchSyntagRelation(pose, questions) → $Oper_1$(questions, pose) ;

searchSemanticPerspective(FL) → perspectiveSemantique retourne la perspective sémantique d'une FL.
Example : searchSemPerspective($Oper_1$) → supportVerb

À partir des fichiers CONLL-X, chaque paire (gov, dep) ayant l'un des 14 types de dépendance a été cherchée dans notre ontologie représentant les relations syntagmatiques extraites du RLF. Nous avons utilisé les lemmes de chaque gov et dep dans la recherche.

---

2. http ://maltparser.org/mco/french_parser/fremalt.html

Une correspondance signifie qu'une paire ayant une relation de dépendance dans une phrase a également une relation syntagmatique dans la langue française et forme peut-être une collocation. Notre recherche dans l'ontologie retourne également la FL reliant la paire. Cela nous permet non seulement d'identifier les collocations, mais aussi d'identifier la FL modélisant la relation.

Pour les résultats positifs, nous conservons les informations suivantes : les formes de surface du gov et du dep, leurs lemmes, la FL qui les connectent dans l'ontologie, l'information sur quel mot est la base et quel est le collocatif et le type de dépendance syntaxique entre les mots dans le texte.

9.2.3 Classification sémantique des collocations

Comme prochain pas, nous recherchons la perspective sémantique (PS) de chaque FL modélisant la collocation.

Puisque les FL peuvent être complexes, c'est-à-dire formées par la combinaison de deux ou plusieurs FL simples, une FL peut avoir plus qu'une PS et, par conséquent, la collocation modélisée par cette FL sera classée dans plus d'un groupe sémantique.

Par exemple, la FL complexe $FinReal_1$ est composée de la FL Fin, dont la PS est action/ événement, sous-classe disparation / disparition existentielle, et par la FL $Real_1$, dont la PS est action/ événement, sous-classe utilisation / opération typique

Ainsi, la collocation abandonner la politique, qui est modélisée par la FL $FinReal_1$ ($FinReal_1$ (politique) = {abandonner}), sera classifiée dans deux groupes sémantiques différents : disparation et utilisation.

Dans le cas où une paire a plus d'une FL, nous choisissons de la classer selon la première FL retournée par la fonction searchSemanticPerspective(gov, dep)

Ci-dessous, voici des exemples de collocations extraites du corpus qui représentent chacune des PS et les FL qui les modélisent :

phase/preparation - déployer ses ailes $PreparReal_1$(ailes) = {déployer};

phase/start - adopter une attitude: $IncepOper_1$(attitude) = {adopter};

phase/continuation - conserver la forme: $ContOper_1$(forme) = {conserver};

action/event - jouer du piano : $Real_1$(piano) = {jouer};

causativity - par politesse: Propt(politesse) = {par};

location - sur le lit : Loc_{in}(lit) = {sur};

opposition - bref délai : AntiMagn(délai) = {bref}. Dans cet exemple, l'opposition n'est pas entre la base (délai) et la valeur (bref). Par contre, c'est une opposition de la relation d'intensification : Magn(délai) = {long} ! AntiMagn(délai) = {bref};

participants - siège vacant: $A_2NonReal_1$(siège) = {vacant};

qualification - politique efficace : Ver(politique) = {efficace};

semant. empty verb - être victime : Pred(victime) = {être};

support verb - courir le risque : $Oper_1$(risque) = {courir};

utilization/form - par avion : Instr(avion) = {par}.

Les PS element/set et equivalence représentent des relations paradigmatiques seulement. Par conséquent, il n'y a pas des collocations classifiées dans ces classes. Exemples des FL et des relations paradigmatiques pour ces classes :

element/set (hyperonymie, méronymie , etc.) - chat/félin : Hyper(chat) = {félin};

equivalence (conversion syntaxique) - acclamation/ acclamer : V_0(acclamation) = {acclamer} ;

equivalence (conversion verbale) - craindre/ e rayer : $Conv_{21}$(craindre) = {e rayer} ;

equivalence (synonymie) - voiture/ automobile : Syn(voiture) = {automobile} ;

9.3 Expérimentations et analyse des résultats

Nous utilisons le corpus EuroSense (Delli Bovi et al., 2017) dans nos expérimentations. EuroSense est un corpus parallèle multilingue contenant des phrases en 21 langues. Il a été automatiquement construit à partir du corpus Europarl (European Parlement)[3], qui est formé par les transcriptions et traductions vers les autres langues o cielles de l'Union européenne des discussions dans le Parlement Européen. Nous l'avons choisi pour trois raisons :

1. Les noms dans le corpus sont liés à leur dé nition dans DBPedia[4]. Cela nous permettra, dans une étude future, de lier nos résultats à des informations linguistiques déjà représentées sur format d'ontologie.
2. Les noms sont désambiguïsés à travers des liens avec DBPedia.
3. Comme le corpus est multilingue, cela nous permettra, dans un travail futur également, de réaliser la même étude en utilisant des phrases dans une autre langue et de la comparer avec nos résultats obtenus pour le français.

Nous avons extrait toutes les phrases en français (environ 1,8 million) contenues dans EuroSense.

---

3. http ://www.statmt.org/europarl/

4. http ://wiki.dbpedia.org/

L'étape suivante est le prétraitement : toutes les phrases sont segmentées (tokenisation) et les tokens sont étiquetés avec leurs parties du discours (POS). Nous utilisons pour ces tâches les logiciels Apache OpenNLP[5] (OpenNLPSegmenter et OpenNlpPosTagger).

Ensuite, la lemmatisation est e ectuée avec DKPRO LanguageToolLemmatizer[6]. MaltParser[7] est utilisé pour l'analyse de dépendance. Finalement, nous utilisons DKPRO pour générer un chier contenant toutes les phrases dans le format CONLL-X.

Selon Canditoet al. (2010), la précision de MaltParser pour le français est d'environ 89%. Ils ont en e et montré que MaltParser est d'environ 1 à 2 points de pourcentage moins précis que BerkeleyParser[8] et MSTParser[9], mais qu'il est environ 12 à 14 fois plus rapide.

Notre choix pour le MaltParser se justi e par sa vitesse, en comparaison aux autres analyseurs syntaxiques, même si cela provoque une perte en précision, perte que nous ne jugeons pas être signi cative.

Nous e ectuons deux types d'analyse des résultats. Dans le premier, nous mesurons la précision dans l'identi cation de collocations par type de dépendance syntaxique. Dans la seconde, nous comptons combien de collocations sont identi- ées par perspective sémantique.

---

5. https ://opennlp.apache.org/
6. https ://dkpro.github.io/
7. http ://www.maltparser.org/
8. http ://www.eecs.berkeley.edu/ petrov/berkeleyParser
9. http ://mstparser.sourceforge.net

9.3.1 Classi cation des collocations par dépendance syntaxique

Le tableau 9.1 montre les collocations identi ées par dépendance syntaxique. Nous avons extrait au total 43 629 candidates à collocation et 33 273 ont été identi és comme de vraies collocations. Cela nous donne une précision générale de 76,2% : nombre de vraies collocations / nombres de candidates extraites automatiquement.

Pour chaque type de dépendance, nous avons le total des candidats extraits, le nombre total de vraies collocations et la précision (total de vraies collocations / total des candidats). Comme il n'existe pas un corpus annoté avec des collocations et des FL pour le français, nous avons calculé la précision manuellement, sur tous les candidats à collocation extraits.

À l'exception de la dépendance _arg_, qui n'a produit que sept collocations, nous avons obtenu la plus haute précision avec les paires ayant les dépendances _obj_ (96,9%), _a_obj_ (98,3%) et _p_obj_ (95,6%). Autrement dit, les relations où le dépendant est l'objet du gouverneur.

Le travail le plus similaire au nôtre est celui de Garcia _et al._ (2017). Ils utilisent des corpus parallèles en trois langues (anglais, portugais et espagnol) et des vecteurs de mots pour faire l'extraction des collocations bilingues (es-pt, es-en et pt-en). La di érence est qu'ils n'utilisent que trois dépendances : _amod_ (adjective-nom), _nmod_ (nom-nom) et _vobj_ (verbe-objet). Ces dépendances sont moins susceptibles de produire des erreurs, car le gouverneur et le dépendant sont adjacents.

Les précisions moyennes obtenues pour les trois paires de langues par Garcia _et al._ (2017) étaient : 91,8% pour _amod_, 90,6% pour _nmod_ et 86,2% pour _vobj_.

À notre connaissance, pour la langue française, il n'y a pas d'autres travaux sur l'extraction de collocations qui utilisent la dépendance syntaxique ou le RLF.

Tableau 9.1: Précision pour l'extraction des collocations par dépendance syntaxique

dépendance	nr. candidats	nr. vraie coll.	précision
mod	20 625	14 240	0,690
dep	14 015	11 532	0,823
obj	4 869	4 720	0,969
suj	1 249	688	0,551
mod_rel	1 179	888	0,753
coord	605	442	0,731
ats	400	346	0,865
a_obj	300	295	0,983
dep_coord	246	13	0,053
p_obj	90	86	0,956
aux_tps	37	15	0,405
arg	7	7	1,000
aux_caus	7	1	0,143
de_obj	0	0	0
Total	43 629	33 273	0,763

En général, nous nous attendions à obtenir une bonne précision pour tous les types de dépendances, puisque chaque candidat a été comparé aux collocations représentées dans notre ontologie basée sur le RLF qui a été construit manuellement par des lexicographes.

Cependant, nous avons obtenu des faux positifs en raison de quelques erreurs dans

l'analyse syntaxique. Les plus fréquentes ont été :

- Des erreurs liées au verbe *être*. Par exemple, dans la phrase *on est pêcheur de père en fils*, le verbe *est* n'est pas syntaxiquement dépendant de *père*. Cependant, l'analyseur a trouvé une dépendance entre eux. Et puisque *être père* est une collocation ($Oper_{12}$ (père) = {être}), nous avons obtenu un faux positif ;
- Des erreurs liées au verbe *avoir*. Par exemple, dans la phrase *...transport des animaux vivants ait été décidée, même si nous aurions préféré...*, l'analyseur syntaxique a trouvé une dépendance *mod* entre *animaux* et *aurions*. Il est considéré comme une collocation car, dans le RLF, il y a la relation suivante : $Real_1$ ($animal_{I:2}$) = {$avoir_{I:1}$} ;
- Des erreurs avec le verbe *pouvoir*. Dans plusieurs phrases, ce verbe a été considéré comme un nom, et nous avons eu des faux positifs parce que, par exemple, *détenir le pouvoir* est une collocation ;
- La conjonction *car* a été confondue avec le nom *car*, qui apparaît de manière incorrecte comme collocation dans certaines expressions comme *dans le car* et *conduire le car*. Dans les paires ayant la dépendance *dep_coord*, *car* était le candidat le plus courant et cela explique pourquoi les candidats de ce groupe ont eu une faible précision.

Par contre, nous avons également obtenu de vrais positifs pour des candidats ayant le verbe *être* comme collocatif, par exemple :

être victime : Pred(victime) = {être} - (verbe sémantiquement vide)

être similaire : Pred(similaire) = {être} - (verbe sémantiquement vide)

Les trois premiers types d'erreurs correspondent à 35,7% de toutes les erreurs, dans la proportion suivante :

des erreurs avec pouvoir : 35,1% ;

des erreurs avec avoir : 31,1% ;

des erreurs avec être : 29,5%.

Les erreurs avec car représentent 3,9%, tandis que le nombre des autres erreurs est de 0,4% (43/10 356).

Cette distribution d'erreurs démontre qu'il serait facile de diminuer le taux d'erreurs en concentrant nos efforts sur le traitement des dépendances entre des lexemes spécifiques ou en faisant une analyse plus détaillée des relations syntagmatiques du RLF où ces lexemes apparaissent.

9.3.2 Collocations classifiées par perspectives sémantiques

Les 33 273 collocations identifiées ont été classifiées par la perspective sémantique (PS) de leurs FL. Comme certaines collocations sont modélisées par des FL complexes, elles peuvent être classifiées en plus qu'une PS, ce qui donne 35 243 instances différentes des classifications par PS.

Le tableau 9.2 montre le nombre de collocations identifiées, classifiées par classe de PS. Pour les PS actionEvent, phaseAspect, qualification et equivalence, nous avons également identifié des collocations par leurs sous-classes.

La PS la plus fréquente pour les collocations était qualification (33,9%), verbe support (24,4%), localisation (17,9%) et action/évènement (9,7%).

Les collocations classifiées dans le groupe qualification ont été presque équitablement divisées entre les sous-groupes intensité et jugement. Comme exemples des collocations appartenant à ce groupe, citons :

grièvement blessés Magn(blesser) = {grièvement} - (PS = intensité)

politique cohérente: Ver(politique) = {cohérente} - (PS = jugement)

Parmi les collocations classifiées comme verbe support, les plus courantes sont celles ayant des FL (simples et complexes) appartenant à la famille $Oper_1$, par exemple :

provoquent le sommeil: $CausIncepOper_1$(sommeil) = {provoquer}

poser une question $Oper_1$(question) = {poser}

Presque toutes les collocations classifiées dans la classe localisation sont modélisées par la FL Loc_{in}. Voici des exemples de collocations appartenant à ce groupe :

dans le pays: Loc_{in} (pays) = {dans}

en semaine: Loc_{in}^{time} (semaine) = {en}

Pour les collocations classifiées comme action/évènement, les sous-groupes les plus communs ont été utilisation/opération typique et création.

instaure la politique: $Caus_1Func_0$(politique) = {instaurer} - (PS = création)

mettre l'accent : $Real_1$(accent) = {mettre} - (PS = utilisation/opération typique)

Ce type d'analyse et de classification par groupe sémantique pourrait être utile, par exemple, pour les applications liées à la sémantique distributionnelle :

Analyse de sentiment : l'identification de collocations dans le groupe sémantique qualification dans une phrase peut être utile pour identifier le sentiment exprimé. Le sentiment sera positif si la collocation est modélisée par les FL Bon ou Ver, et négatif si la collocation est modélisée par les FL AntiBon ou AntiVer ;

Classi cation du texte : la présence d'une collocation spéci que dans un groupe sémantique spéci que peut aider à identi er le sujet d'une phrase ou d'un texte.

Nous soulignons que, à notre connaissance, il n'y a pas d'autres travaux sur la classi cation sémantique des collocations en fonctions des FL qui les modélisent.

Tableau 9.2: Nombre de collocations identifiées par perspective sémantique

perspective sémantique	n. coll.	exemple	fonction lexicale
qualification	11 983		
intensity	5 988	très grave	Magn(grave) = {très}
judgment	5 940	aller bien	Bon(aller) = {bien}
inten+judg	55	trop rapide	AntiBon+Magn(rapide) = {trop}
supportVerb	8 620	donner un coup	$Oper_1$(coup) = {donner}
location	6 332	dans le pays	Loc_{in}(pays) = {dans}
actionEvent	3 428		
utilizationTypOper	2 846	mettre l'accent	$Real_1$(mettre) = {accent}
creation	580	faire voler	Caus(voler) = {faire}
disparExistCease	121	l'avion atterrit	$FinFact_0$(avion) = {atterrir}
imminence	5	la tempête vient	$ProxFunc_0$(tempête) = {venir}
semanticallyEmptyVerb	1 693	être similaire	Pred(similaire) = {être}
utilizationForm	1 410	à vélo	Instr(vélo) = {à}
phaseAspect	729		
start	632	devenir mère	IncepPred(mère) = {devenir}
continuation	77	rester debout	ContPred(debout) = {rester}
preparation	20	déployer les ailes	$PreparReal_1$(ailes) = {déployer}
semanticOpposite	647	mauvais sens	AntiBon(sens) = {mauvais}
causativity	362	par politesse	Propt(politesse) = {par}
participants (actants)	39	cigarette allumé	A_1Fact_0(cigarette) = {allumer}
Total	35 243		

9.4 Conclusion

Dans ce chapitre, nous avons présenté une méthode basée sur l'analyse des dépendances syntaxiques et sur le RLF en format d'ontologie pour l'identification de collocations extraites d'un corpus en français.

Après le prétraitement, un fichier en format CONLL-X contenant les phrases analysées est généré. Ensuite, les candidats à collocation sont extraits en fonction de 14 différentes relations de dépendance syntaxique.

Pour décider si un candidat est une vraie collocation, il est recherché dans le RLF en format d'ontologie. Cependant, en plus des collocations, nous obtenons également la fonction lexicale reliant la base et le collocatif de chaque collocation et la perspective sémantique de chaque fonction lexicale.

Ensuite, nous calculons la précision de l'identification des collocations pour chaque type de dépendance syntaxique et nous comparons l'ensemble des collocations identifiées par perspective sémantique.

Pour certains types de dépendance, nous avons obtenu une précision supérieure à 95%, et nous avons analysé que, en traitant correctement les problèmes ponctuels, comme certaines erreurs dans l'analyseur syntaxique, nous pouvons obtenir une précision proche de 100%.

Dans nos travaux futurs, nous avons l'intention de combiner notre méthode avec les algorithmes d'apprentissage machine basés sur des prolongements de mots (en anglais word embeddings). L'apprentissage machine fonctionne bien pour les collocations ayant une fréquence supérieure à un certain seuil. Nous croyons que l'utilisation d'un réseau lexical où les relations de collocations (relations syntagmatiques) ont été manuellement annotées par des lexicographes aidera à identifier aussi les collocations moins fréquentes.

Aussi comme travail futur, nous souhaitons étendre cette méthode à un réseau lexical basé sur des fonctions lexicales pour l'anglais (en construction) et à l'appliquer à la traduction de collocations pour la paire de langues anglais-français.

Finalement, nous avons l'intention de créer un corpus français annoté avec des collocations et fonctions lexicales, pour permettre une future évaluation automatique de la précision et du rappel lors de l'identi cation de collocations.

CONCLUSION

Nous avons vu dans la présente thèse le phénomène linguistique appelé collocation, qui est une expression formée par deux parties, la base et le collocatif, dont le collocatif est choisi en dépendance avec la base pour exprimer un sens spécifique.

Les collocations sont nombreuses dans toutes les langues et posent des problèmes aux tâches liées au traitement automatique des langues (TAL), tel que la traduction automatique et la génération de texte.

La plupart des problèmes liés aux collocations dans le domaine du TAL sont sémantiques, puisque les collocations font intervenir des acceptions de vocables dont le sens n'est pas prévisible et dont la combinatoire est capricieuse.

De plus, en dépit de sa fréquence dans les langues et des problèmes qu'elles posent, le traitement informatique des collocations est encore peu développé, en particulier la représentation des collocations.

Les fonctions lexicales (FL) sont un outil linguistique créé pour la représentation du sens existant entre les deux parties d'une collocation ainsi que les informations combinatoires et syntaxiques connectant la base et le collocatif. En plus de représenter le sens prédicatif connectant les collocations, qui est une relation du type syntagmatique, les FL représentent aussi des relations de type paradigmatique entre des unités lexicales.

Des implémentations informatiques ont été proposées pour représenter les FL, la plupart d'entre elles les reliant à des grammaires formelles, tel que HSPG et LFG, ou dans des entrées de dictionnaires. Pourtant, une implémentation des

FL indépendante de la représentation des unités lexicales et des collocations est manquante.

Comme exemple, Heylen et al. (1994) ont suggéré la représentation de collocations dans une grammaire HPSG. Ils proposent que l'information sur une collocation soit placée dans la représentation MAV du mot qui est la base de cette collocation. Pour représenter la collocation *peur bleue*, qui est modélisée par la FL Magn (Magn(*peur*) = {*bleue*}), il faut que le nom de la fonction Magn soit représenté avec le mot *bleue* dans le MAV du mot *peur*.

Les problèmes principaux avec ce type de représentation (et avec la majorité des représentations proposées qui suivent ce patron), sont les suivants :

1. Les informations sur les FL ne sont pas représentées. Dans l'exemple précédent, il n'y a pas d'information sur la fonction Magn : quel est son sens prédicatif ? Comment se combine-t-elle avec d'autres fonctions simples pour former des FL complexes ? Quel est son comportement syntaxique ? etc. ;
2. Bien que plusieurs collocations sont modélisées par une même FL, par exemple Magn, ce type de représentation ne permet pas d'informer que ces collocations sont connectés par un même sens prédicatif ;
3. Il n'est pas possible de partager les informations combinatoires entre des FL qui sont similaires. Par exemple, entre les FL Oper, Func et Labor ou entre les FL Bon et Ver.

Pour les raisons exposées, en plus de représenter les collocations, nous avons proposé dans ce travail de recherche une nouvelle représentation des FL. Une représentation dans laquelle on a des informations sur les FL : informations sémantiques, syntaxiques et combinatoires.

Nous avons choisi les formalismes du Web sémantique, les langages RDF/OWL,

comme formalismes informatiques pour représenter les FL. Ces formalismes sont utilisés pour la création des ontologies informatiques.

Une représentation RDF est un triplet : propriété(ressource) = valeur. D'une manière similaire, la majorité des collocations sont aussi un triplets: sens prédicatif (base) = valeur. Les exceptions sont les collocations formées par des verbes supports, lesquels ne portent pas de sens.

Une autre raison pour utiliser le langage RDF est que les informations représentées forment un graphe orienté : les objets sont des n÷uds et les propriétés connectant des objets forment des arcs. Cela nous semble une manière naturelle d'encoder des informations combinatoires entre des unités lexicales, car le lexique d'une langue peut être vu comme un graphe (ou réseau), où les mots sont des n÷uds et les relations lexicales (relations paradigmatiques et syntagmatiques) sont des arcs.

L'utilisation du langage OWL est justi é par son pouvoir de raisonnement, car il est basé sur une logique formelle, la logique de description.

De plus, les langages RDF et OWL sont parmi les formalismes les plus utilisés pour la représentation des connaissances (Hendler et van Harmelen, 2008). En utilisant ces formalismes, nous pouvons connecter notre modèle de représentation de collocations et de fonctions lexicales à plusieurs autres outils et ressources, qui représentent d'autres informations linguistiques ou d'autres domaines.

Pour que les informations linguistiques et méta-linguistiques (partie du discours, informations morphologiques, syntaxiques, etc.) soient représentées sur le Web sémantique, des ontologies méta-linguistiques ont été développées. Ces ontologies ont été évolués jusqu'au développement de lemon, qui est devenu un standard de facto . Pour cette raison, le modèle que nous avons développé pour représenter les FL et les collocations est compatible avec lemon.

Finalement, comme application de notre modèle, nous avons transformé un réseau lexical, le Réseau Lexical du Français, de son format initial de base de données relationnelle vers un format d'ontologie, compatible avec d'autres ontologies basées sur RDF et OWL.

1. Contributions

Les contributions de la présente thèse sont les suivantes :

1. création d'une ontologie informatique pour la représentation des informations des fonctions lexicales : informations sémantiques, syntaxiques et combinatoires ;
2. création d'une ontologie informatique pour la représentation des collocations ;
3. création d'une ontologie informatique pour la représentation d'une classi - cation sémantique des fonctions lexicales ;
4. représentation d'environ 600 fonction lexicales, dont environ 100 fonctions lexicales simples et environ 500 fonctions lexicales complexes ;
5. représentation d'environ 54 000 relations lexicales, dont environ 46 000 paradigmatiques et environ 8 000 syntagmatiques (collocations) ;
6. transformation du Réseau lexical du français vers un format d'ontologie.

La création des ontologies, la représentation des fonctions lexicales et des collocations ont été faites d'une manière semi-automatique.

Notre modèle de représentation des fonctions lexicales et des collocations est unique et original. Nous croyons qu'il contribuera directement aux domaines de la lexicographie et du traitement automatique des langues, comme par exemple, dans la construction des ontologies lexicales et dans l'extraction de l'information

lexicale à partir du texte.

De la même manière que les réseaux lexicaux comme WordNet ont contribué à la recherche linguistique, psycholinguistique et informatique, nous croyons que l'utilisation de notre implémentation des relations lexicales extraites du RLF dans un format compatible avec les formalismes du Web sémantique sera utile dans plusieurs tâches, comme la désambiguïsation et la génération de texte.

2. Perspectives

Nous présentons dans cette section des perspectives sur l'utilisation des réseaux lexicaux encodés avec notre modèle pour la traduction des collocations et la désambiguïsation des mots combinée avec la sémantique distributionnelle.

2.1. Encodage du Réseau lexical de l'anglais

Le Réseau lexical de l'anglais (RL-en) est présentement en développement à l'ATILF [10]. Nous envisageons de le transformer en ontologie en utilisant notre modèle.

2.2. Traduction automatique de collocations

Une fois le RL-en encodé avec notre modèle, nous pouvons construire un système de traduction automatique de collocations (anglais vers le français et français vers l'anglais). Comme les deux réseaux seront encodés avec le même modèle de représentation de collocations, la tâche d'appariement des collocations dans les deux langues sera plus facile.

---

10. http ://www.atilf.fr/

2.3. Utilisation du RLF avec la sémantique distributionnelle

L'apprentissage automatique, surtout l'apprentissage profond, produit présentement les résultats les plus satisfaisants pour certaines tâches liées au TAL, comme la traduction automatique et la désambiguïsation lexicale.

Cette amélioration de performance a été accélérée spécialement après le développement deWordToVec[11] (Mikolov et al., 2013), qui permet de projeter les termes (contenus dans une fenêtre sémantique dé nie) d'une langue étudiée dans un espace de représentation vectorielle pour qu'ils puissent être utilisés comme entrées d'un réseau de neurones.

Cependant, la combinaison de l'apprentissage profond et les informations lexicales déjà encodées dans des bases de connaissances linguistiques et des réseaux lexicaux soulève des questions quant aux gains de performance. Par exemple, il faut transformer l'information sur forme de réseaux vers une forme de matrice avant que l'information sois utilisée dans un réseau des neurones arti ciel.

Bordes et al. (2011) proposent une architecture pour transformer l'information contenue dans un graphe lexical vers les vecteurs de mots. Ils appliquent cet architecture à WordNet et à Freebase[12].

Johansson et Piña (2015) présentent aussi une méthode pour la transformation de l'information dans un réseau lexical vers les vecteurs de mots. La di érence dans leur approche est que des mots polysémiques sont représentés par une combinaison de vecteurs de sens.

Comme travail futur, nous envisageons l'application d'une méthode similaire à

---

11. https ://code.google.com/archive/p/word2vec/

12. https ://developers.google.com/freebase/

celle de Johansson et Piña (2015) pour la représentation des relations sémantiques présentes dans le RLF comme des vecteurs de mots et de sens, pour une tâche de désambiguïsation lexicale. Nous pourrons, ensuite, utiliser une évaluation SemEval[13] pour comparer notre implémentation et un système de désambiguïsation qui utilise des réseaux comme WordNet et DBPedia.

Comme avantages d'utiliser le RLF avec la sémantique distributionnelle, nous citons que, premièrement, il est déjà désambiguïsé. Deuxièmement, il contient des relations syntagmatiques qui sont absentes des réseaux utilisés par Bordes et al. (2011) et par Johansson et Piña (2015), par exemple. Une quantité plus large et des relations lexicales plus variées impliquent plus d'information sémantique pour désambiguïser des mots polysémiques dans un texte.

13. http ://alt.qcri.org/semeval2016/

ANNEXE A

PUBLICATIONS

1. Articles publiés, soumis et acceptés

a. Articles de revue

Billal Belainine, Alexsandro Fonseca , Fatiha Sadat and Hakim Lounis (2018, to appear). Semi-supervised learning and Social Media Text Analysis towards multi-labeling categorization (version étendue de l'article BIGDATA-2017). International Journal of Data Mining Science. - http ://www.ijdat.org/index.php/ijdat

b. Article publié comme chapitre de livre

Alexsandro Fonseca , Fatiha Sadat and François Lareau (2017). Combining dependency parsing and a lexical network based on lexical functions for the identi cation of collocations. Ruslan Mitkov (Ed.), Computational and Corpus-based Phraseology. Lecture Notes in Arti cial Intelligence, Volume 10596, pp. 447-461 November 2017, Springer International Publishing. ISBN 978-3-319-69805-2_31.

c. Articles des conférences et workshops

Alexsandro Fonseca , Fatiha Sadat and François Lareau (2018). Retrieving Information from the French Lexical Network in RDF/OWL Format. Dans The 11th International Conference on Language Resources and Evaluation (LREC-2018), pp. 4408-4412, Miyazaki, Japan. May 7-12th, 2018.

Jean-Marie Poulin, Alexandre Blondin Massé et Alexsandro Fonseca . Strategies for Learning Lexemes E ciently : A Graph-Based Approach (2018). Dans The Tenth Intl. Conf. on Advanced Cognitive Technologies and Applications, Think Mind Digital Library (INRIA), pp. 18-23, Barcelona, Spain, Feb., 2018.

Billal Belainine, Alexsandro Fonseca , Fatiha Sadat et Hakim Lounis (2017). Semi-supervised learning and Social Media Text Analysis towards multi-labeling categorization.2017 IEEE International Conference on Big Data(Big Data 2017), pp. 1907-1916, Boston, MA, USA, December 11-14th, 2017.

Alexsandro Fonseca , Fatiha Sadat et François Lareau (2016). Lexfom : a Lexical functions ontology model.In Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex-V), COLING-2016, pp. 145-155, Osaka, Japan, December 12th 2016.

Billal Belainine, Alexsandro Fonseca et Fatiha Sadat (2016). Named Entity Recognition and Hashtag Decomposition to Improve the Classi cation of Tweets. In Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT), COLING 2016, pp. 64-73, Osaka, Japan, December 11th 2016.

Billal Belainine, Alexsandro Fonseca et Fatiha Sadat (2016). E cient Natural Language Pre-processing for Analyzing Large Data Sets. In Proceedings of the Workshop on Big Data and Natural Language Processing (BIG-NLP), IEEE BigData-2016, pp. 3864-3871, Washington D.C., USA, December 5th 2016.

Alexsandro Fonseca , Fatiha Sadat et François Lareau (2016). A Lexical Ontology to Represent Lexical Functions.In Proceedings of the 2nd Workshop on Language and Ontology (LangOnto2), LREC-2016, pp. 69-73, Portoroz, Slovenia, May 23rd 2016.

Alexsandro Fonseca et Fatiha Sadat (2014). A Comparative Study of Di erent Classi cation Methods for the Identi cation of Brazilian Portuguese Multiword Expressions.In Proceedings of the First Workshop on Computational Approaches to Compound Analysis (ComAComA), COLING-2014, pp. 53 62, Dublin, Ireland, August 24th 2014.

Alexsandro Fonseca , Fatiha Sadat et Alexandre Blondin Massé (2014). Identifying Portuguese Multiword Expressions using Di erent Classi cation Algorithms A Comparative Analysis. In Proceedings of the 4th International Workshop on Computational Terminology (CompuTerm), COLING-2014, pp. 104 113, Dublin, Ireland, August 23rd 2014.

2. Articles à soumettre

a. Articles de revue

Alexsandro Fonseca , Fatiha Sadat et François Lareau. Lexical Functions in Semantic Web Formalisms for Collocation Representation. International Journal of Lexicography -https://academic.oup.com/ijl

Jean-Marie Poulin, Alexandre Blondin Massé et Alexsandro Fonseca . Strategies for Learning Lexemes E ciently : A Graph-Based Approach (version étendue de l'article de la conférence COGNITIVE-2018) (2018, pré-accepté). International Journal On Advances in Intelligent Systems.

3. Présentations

Billal Belainine, Alexsandro Fonseca , Fatiha Sadat et Hakim Lounis (2017, présentation par a che). Multi-label Semi-Supervised Classi cation of Social Media Documents applied via Di erent Machine Learning Algorithms.Première édition du Symposium IA Montréal, Montréal, Canada, le 26 sep, 2017.

Alexsandro Fonseca (2017). Lexfom : une ontologie lexicale pour la représentation des fonctions lexicales. Présenté au Séminaire RALI-OLST (Recherche appliquée en linguistique informatique - Observatoire de linguistique Sens-Texte), Université de Montréal, le 22 février 2017.
http://rali.iro.umontreal.ca/rali/?q=fr/node/1222/show/3727

ANNEXE B

FONCTIONS LEXICALES IMPLÉMENTÉS DANS LEXFOM

Voici la liste des fonctions lexicales implémentées dans notre modèle lexfom. Elles sont divisées en deux groupes : les paradigmatiques et les syntagmatiques. Chacun de ces deux groupes est divisé par famille de fonctions lexicales (encodée par le module lam).

La division par famille suit la division implémentée par le Réseau Lexical du Français.

1. Fonctions lexicales paradigmatiques

Famille A_0

A_0 A_0_inter A_0_less A_0_more A_0Mult A_0Real_1

Famille A_1

A_1 A_1_inter A_1_less A_1_more $A_1\_or\_A_0$ $A_1\_sl\__2$
$A_1\_sl\__2\_less$ $A_1\_sl\__2Perf$ $A_1\_sl\__3$ $A_1AntiBonReal_1$ A_1Fact_0
$A_1Magn\ tempReal_1$ A_1Non $A_1NonFact_0$ $A_1NonFact_2$ $A_1NonReal_1$
A_1Perf A_1Pred $A_1\ prime$ A_1Real_{xt} A_1Real_1 A_1Real_2
$Magn\_plus\_A_1$ $Magn\ quant\_plus\_A_1$ S_0A_1

Famille A_2

A_2 A_2_inter $A_2\_sl\__3$ $A_2\_sl\__3Perf$ A_2Fact_1 A_2Fact_2
$A_2LiquFunc_0$ A_2Manif $A_2NonReal_1$ A_2Perf $A_2PerfPrepar_1$
A_2Real_1

Famille A_3

A_3 $A_3LiquFunc_0$ A_3Perf

Famille A_4

A_4

Famille $Able_1$

$Able_1$ $Able_1\_or\_A_1$ $Able_1Fact_0$

Famille $Able_2$

$Able_2$ $Able_2\_or\_Able_1$ $Able_2Real_{xt}$

Famille $Able_3$

$Able_3$ $Able_3\_or\_Able_1$ $Able_3\_or\_Able_2$

Famille $Able_4$

$Able_4$ $Able_4\_or\_Able_1$

Famille Adv_0

Adv_0 Adv_0_inter Adv_0_more $Adv_0\_or\_A_0$

Famille Adv_1

Adv_1 $Adv_1\_or\_A_1$ $Adv_1Real_2^{III}$

Famille Adv_2

Adv_2 $Adv_2\_or\_A_2$

Famille Anti

Anti Anti_inter Anti_less Anti_more

Famille AntiAble

$AntiAble_1Fact_0$ $AntiAble_2$ $AntiAble_2\_or\_AntiAble_1$ $AntiAble_2Real_{xt}^{II}$

Famille Cap

Cap

Famille Contr

Contr Fem Masc

Famille Conv

$Conv_2$ $Conv_{21}$ $Conv_{21}\_inter$ $Conv_{21}\_less$ $Conv_{21}\_more$ $Conv_{213}$ $Conv_{23}$ $Conv_{312}$ $Conv_{32}$ $Conv_{321}$ $Conv_{423}$ $S_2\_et\_Conv_{21}$

Famille Culm

Culm

Famille Degrad

Degrad CausDegrad

Famille Denouveau

Denouveau

Famille Equip

Equip SingEquip

Famille Figur

Figur

Famille Gener

Gener

Famille Mult

Holo IncepPredMult Mult Mult _inter Mult _less Mult_more

Famille Pred

Pred Pred_inter $Caus_{xt}$Pred $Caus_1$_sl_$_2$Pred $Caus_1$DenouveauPred $Caus_1$Pred $Caus_2$Pred CausDenouveauPred CausPred ContPred $Conv_{21}$Pred IncepPred $Liqu_1$Pred LiquPred

Famille $Qual_1$

$Qual_1$

Famille $Qual_2$

$Qual_2$

Famille $Qual_3$

$Qual_3$

Famille $Result_1$

$Result_1$

Famille $Result_2$

$Result_2$

Famille $Result_3$

$Result_3$

Famille S_0

$Magn_1$ quant_plus_S_0 S_0 S_0 usual S_0_inter S_0_less S_0_more S_0Conv_{21} S_0Conv_{213} S_0Pred S_0Pred_inter S_0Pred_less S_1_or_S_0 S_2_or_S_0 S_3_et_S_0 S_4_et_S_0 $SingS_0$ Sres_et_S_0 Sres_or_S_0

Famille S_1

Magn_plus_S_1 Mult_et_S_1 $MultS_1$ $MultS_1$_more $MultS_1$_sl_2 $MultS_1$ prototype S_1 S_1:$_1$ S_1:$_1$ prototype S_1:$_2$_more S_1:$_2$ prototype S_1_inter S_1_inter usual S_1_less S_1_less usual S_1_more S_1_more usual S_1_pl_2 S_1_pl_2_more$Pred$ S_1_sl_2 S_1_sl_2Perf S_1_sl_2Pred S_1_sl_2 prototype S_1_sl_2 usual S_1Perf S_1Pred S_1 prototype S_1 prototype$Pred$ S_1 usual S_1 usual$Pred$ Sing_et_S_1 Sing_et_S_1_sl_2 $SingS_1$ $SingS_1$_more

Famille S_2

Cap_et_S_2 $MultS_2$ S_2 S_2:$_1$ S_2:$_2$ S_2:$_2$_more S_2:$_2$ prototype S_2_inter S_2_more S_2_sl_3 S_2_sl_3Perf S_2_sl_4 prototype

S_2Perf S_2 prototype S_2 usual Sinstr_et_S_2 Sloc_et_S_2 Sloc_et_S_2 prototype Smed_et_S_2 Sres_et_S_2

Famille S_3

MultS_3 S_3 S_3_inter S_3_more S_3_or_S_1 S_3_sl_5 S_3 prototype S_3 usual SingS_2 Sinstr_et_S_3 Sinstr_et_S_3 prototype Sloc_et_S_3 Smed_et_S_3 prototype Smod_et_S_3

Famille S_4

S_4 S_4Perf S_4 prototype Sinstr_et_S_4 Sinstr_et_S_4 prototype Sloc_et_S_4 Smed_et_S_4 prototype

Famille Sing

Mero MeroSing Sing Sing_inter Sing_less Sing_more

Famille Sinstr

Sinstr SinstrCausFunc$_0$ SinstrReal$_1$ SinstrReal$_1$_ap_$_1$ SinstrReal$_1^{\mathrm{I}}$ SinstrReal$_1^{\mathrm{II}}$

Famille Sloc

Sloc SlocCausFunc$_0$ SlocFact$_0$ SlocIncepFunc$_0$ SlocNonFact$_0$ Sloc temp Sloc tempFact$_0$

Famille Smed

Smed

Famille Smod

Smod

Famille Sres

Sres

Famille Syn

Anc (Syn) CapMult Hypo Syn Syn_inter Syn_less
Syn_less sex Syn_more Syn_more sex

Famille V_0

Enun EnunCaus$_1$Pred V_0 V_0_inter V_0_less V_0_more V_0Conv$_{21}$
V_0Conv$_{213}$ V_0Conv$_{413}$

2. Fonctions lexicales syntagmatiques

Famille AntiBon

AntiBon AntiBon_plus_AntiMagn AntiBon_plus_AntiMagn_largeur
AntiBon_plus_AntiMagn_taille AntiBon_plus_AntiMagn_vitesse
AntiBon_plus_AntiVer AntiBon_plus_Magn
AntiBon_plus_Magn_quant AntiBon_plus_Magn_temp
AntiBon_plus_Ver AntiBon$_1$ AntiBon$_2$ AntiBonA$_1$Fact$_0$
AntiBonFact$_0$ AntiBonFact$_1$ AntiBonFact$_2$ AntiBonLiqu$_1$Fact$_2$
AntiBonReal$_{xt}$ AntiBonReal$_1$ AntiBonReal$_1$_usual AntiBonS$_2$

Famille AntiMagn

AntiMagn AntiMagn_epaisseur AntiMagn_hauteur AntiMagn_largeur
AntiMagn_largeur_plus_Fact$_1$ AntiMagn_longueur AntiMagn_plus_A$_1$
AntiMagn_plus_S$_1$ AntiMagn_puissance AntiMagn_taille
AntiMagn_vitesse AntiMagn$_1$ quant AntiMagn$_2$ quant
AntiMagn quant AntiMagn quant_plus_Magn_vitesse
AntiMagn quantS$_2$ AntiMagn temp

Famille AntiVer

AntiVer AntiVer_finalite AntiVer_utilisation CausPredAntiVer

Famille Bon

Bon Bon_inter Bon_plus_A$_1$ Bon_plus_AntiMagn Bon_plus_Magn
Bon_plus_Magn quant Bon_plus_Ver CausPredBon

Famille Caus

Caus Caus$_1$_sl_2 CausConv$_{21}$ CausConv$_{231}$ EnunCaus S$_0$Caus

Famille Cont

Cont

Famille CausObstr

CausObstr

Famille Epit

Epit Redun Redun_more

Famille $Fact_0$

$AntiFact_0$ $AntiPermFact_0$ $BonFact_0$ $Caus_1DenouveauFact_0$ $Caus_1Fact_0$ $Caus_1MinusFact_0$ $Caus_1PlusFact_0$ $CausContFact_0$ $CausFact_0$ $Fact_0$ $Fact_0^{I}$ $Fact_0^{II}$ $FinFact_0$ $IncepFact_0$ $Liqu_1Fact_0$ $Liqu_3Fact_0$ $LiquFact_0$ $MagnFact_0$ $Mult\_plus\_Fact_0$ $NonFact_0$ $PermFact_0$ $ProxFact_0$ $S_0AntiBonFact_0$ S_0Fact_0 $S_0IncepFact_0$ $S_0SingFact_0$ $S_1CausFact_0$ $S_1\ primeFact_0$ $SresFact_0$ $SresFact_0^{III}$ $SresPerfFact_0$

Famille $Fact_1$

$CausFact_1$ $Fact_1$ $Liqu_3Fact_1$ $LiquFact_1$ $Magn\ quant\_plus\_A_2Fact_1$ $PreparFact_1$

Famille $Fact_2$

$Fact_2$ $Fact_2\_sl\_3$ $Fact_{23}$ $Fact_2^{I}$ $Fact_2^{II}$ $Fact_2\ usual$ $Magn_2\ quant\_plus\_A_1Fact_2$ $SresFact_2$ S_0Fact_2 $Prepar_1Fact_2$

Famille $Fact_3$

$Fact_3$

Famille $Fact_{xt}$

$Fact_{xt}$

Famille Fin

Fin

Famille $Func_0$

$AntiVerCausFunc_0$ $Caus_1Func_0$ $Caus_2Func_0$ $CausContFunc_0$ $CausDenouveauFunc_0$ $CausFunc_0$ $ContFunc_0$ $DenouveauFunc_0$ $EnunIncepFunc_0$ $EnunNonPermFunc_0$ $FinFunc_0$ $Func_0$ $IncepFunc_0$ $Liqu_1Func_0$ $LiquFunc_0$ $Magn\_plus\_CausFunc_0$ $Magn\ quant\_plus\_Func_0$ $NonPerm_1Func_0$ $Perm_2Func_0$ $ProxFunc_0$ $S_0Caus_2Func_0$ $S_0FinFunc_0$ $S_1CausFunc_0$ $S_1\ usualCausFunc_0$ $S_3LiquFunc_0$ $SlocFinFunc_0$ $VerCausFunc_0$

Famille $Func_1$

$Caus_2Func_1$ $CausFunc_1$ $Func_1$ $Func_1\_pl\_2$ $Func_{12}$ $LiquFunc_1$ $IncepFunc_1$ $Liqu_2Func_1$ $Magn\_plus\_Func_1$ $NonPerm_2Func_1$

Famille $Func_2$

$Caus_2Func_2$ $Caus_1Func_2$ $CausFunc_2$ $Func_2$ $Func_{23}$ $IncepFunc_2$ $LiquFunc_2$ $Magn\_plus\_Func_2$

Famille $Func_3$

$Func_3$

Famille Germ

Germ

Famille Incep

Incep

Famille Instr

Instr

Famille Involv

Involv A_1Conv_{21}Involv $Adv_1AntiAble_2$Involv AntiBon _ plus_ A_1Involv AntiBonInvolv $Caus_1$AntiBonInvolv $Caus_1$Involv $Conv_{21}$Involv $NonConv_{21}$Involv SinstrAntiBonInvolv SresAntiBonInvolv

Famille $Labor_{12}$

$Labor_{12}$

Famille $Labor_{21}$

$Caus_2Labor_{21}$ $Labor_{21}$

Famille $Labreal_{12}$

$AntiLabreal_{12}$ $Labreal_{12}$ $Labreal_{123}$ $Labreal_{12}^{I}$ $Labreal_{12}^{II}$ $S_0Labreal_{12}$

Famille $Labreal_{13}$

$Labreal_{13}$

Famille $Labreal_{21}$

$Labreal_{21}$

Famille $Labreal_{23}$

$Labreal_{23}$

Famille $Labreal_{xt}$

$Labreal_{1xt}$

Famille Liqu

Liqu

Famille Loc

Loc_ab Loc_ad Loc_in Loc_in temps

Famille Magn

Magn Magn_comportement Magn_effet Magn_epaisseur Magn_hauteur Magn_largeur Magn_longueur Magn_puissance Magn_taille Magn_vitesse Magn_vitesse_plus_Magn quant $Magn_1$ quant $Magn_2$ $Magn_2$ quant Magn quant Magn temp

Famille Manif

Manif NonPerm $_1$Manif

Famille Minus

CausPredMinus IncepPredMinus

Famille $Oper_1$

$CausIncepOper_1$ $CausOper_1$ $CausOper_{12}$ $Caus_1NonOper_1$ $Caus_1Oper_1$ $Caus_2Oper_1$ $Caus_3Oper_1$ $CausFinOper_1$ $ContOper_1$ $DenouveauOper_1$ $EnunCausOper_1$ $FinOper_1$ $IncepOper_1$ $IncepOper_{12}$ $Liqu_1Oper_1$ $LiquOper_1$ $Magn\_plus\_Oper_1$ $Magn\ quant\_plus\_Oper_1$

Magn quant_plus_$Oper_{12}$ $NonOper_1$ Oper_si$_1$ $Oper_1$ $Oper_{1}$_ap_1 $Oper_{1}$_sl_2 $Oper_{1}$_sl_3 $Oper_{12}$ $Oper_{123}$ $Oper_1$ prime $Oper_1$ usual $Prepar_{xt}$ $Oper_1$ $S_1CausOper_1$ S_1 prime$Oper_1$

Famille $Oper_2$

$CausOper_2$ $Caus_1Oper_2$ $Caus_2Oper_2$ $ContOper_2$ $FinOper_2$ $IncepOper_2$ Magn_plus_$Oper_{21}$ $Oper_2$ $Oper_{2}$_sl_3 $Oper_{21}$ $Oper_{213}$ $Oper_{23}$

Famille $Oper_3$

$Oper_3$ $Oper_{31}$ $Oper_{32}$

Famille $Oper_{41}$

$Oper_{41}$

Famille Plus

$CausPredPlus$ $IncepPredPlus$ $IncepPredPlus$ taille

Famille Prepar

$A_2Prepar_1$ $AntiPrepar$ $Prepar$ $Prepar_1$ $Prepar^{I}$ $Prepar^{II}$ $Prepar_1Fact_0$ $PreparFact_0$ $PreparFact_0^{I}$ $PreparFact_0^{II}$

Famille Propt

Propt

Famille $Real_1$

$AntiPrepar_1Real_1$ $AntiReal_1$ $AntiVerReal_1$ $BonReal_1$ $BonReal_2$ $Caus_1PlusReal_1$ $Caus_1Real_1$ $CausReal_1$ $ContReal_1$ $Enun_1Real_1$ $FinReal_1$ $IncepReal_1$ $Liqu_1Real_1$ $Liqu_1Real_1\_ap\_1$ $PerfReal_1$ $Real_1$ $Real_1\_ap\_1^{I}$ $Real_1\_ap\_1^{II}$ $Real_1\_ap\_1$ $Real_1\_sl\_2$ $Real_{12}$ $Real_{13}$ $Real_1^{I}$ $Real_1^{II}$ $Real_1^{III}$ $Real_1$ usual $Magn\_plus\_Real_1$ $Magn_2$ $quantCaus_1Real_1$ $Prepar_1Real_1$ $PreparLiqu_1Real_1$ $PreparReal_1$ $PreparReal_1^{I}$ $PreparReal_1^{II}$ S_0Real_1 S_0Real_{12} $S_0Real_1^{II}$ S_0Real_1 usual $S_0SingReal_1$ S_1Real_1 $SinstrPreparReal_1$ $SlocReal_1^{III}$ $SlocReal_1$ usual $SmedReal_1$ $SresCaus_1Real_1$ $SresReal_1$

Famille $Real_2$

$AntiReal_2$ $AntiReal_{21}$ $AntiReal_2^{I}$ $IncepReal_2$ $Caus_1Real_2$ $FinReal_2$ $Liqu_1Real_2$ $Real_2$ $Real_{21}$ $Real_2^{I}$ $Real_2^{II}$ $Real_2^{III}$ $Real_2$ usual $Prepar_1Real_2$ S_0Real_2

Famille $Real_3$

$LiquReal_3$ $PreparReal_3$ $Real_3$ $Real_{31}$

Famille $Real_4$

$Real_4$

Famille $Real_{xt}$

$AntiReal_{xt}$ $AntiReal_{xt}^{III}$ $Caus_1Real_{xt}$ $PreparReal_{xt}$ $Real_{xt}$ $Real_{xt}^{I}$ $Real_{xt}^{II}$ $Real_{xt}^{III}$ S_0Real_{xt} S_1Real_{xt} $S_1Real_{xt}^{I\_sl\_II}$ $SresReal_{xt}$

Famille Son

EnunSon S_0Son Son

Famille V er

CausP redV er ContP redV er SinstrCausP redV er V er V erfinalite

ANNEXE C

PROCESSUS DE DÉVELOPPEMENT DE LEXFOM ET DE TRANSFORMATION DU RLF

Dans cette annexe, nous expliquons le développement des modules de notre ontologie et la transformation du Réseau lexical du français (RLF) dans un format compatible avec le Web sémantique.

1. Développement des modules de lexfom

Chaque module a été construit en utilisant l'outil Protégé[1], version 5.2.0. La section 1.1 montre un exemple des fenêtres du module lexical functions representation (lfrep) dans Protégé. Dans les figures, les cercles représentent des classes et les losanges représentent des instances de chaque classe.

À la section 1.2, nous expliquons le développement des modules de l'ontologie lexfom et à la section 1.3, le processus de transformation du RLF.

1.1. Module lexical functions representation (lfrep)

La figure C.2 montre les classes du module lfrep.

Les figures C.3 à C.7 montrent les instances des classes du module lfrep.

1.2. Développement des modules de l'ontologie lexfom

Les quatre modules de notre modèle ont été développés manuellement, en utilisant Protégé. Les inspirations pour chaque module étaient les suivantes :

---

1. https ://protege.stanford.edu/

Figure C.2: Classes du module *lexical function representation*

(a) Classe *circunstanceSpecification* (b) Classe *dimensionMode*

Figure C.3: Instances des classes du module *lfrep*

(a) Classe equivalenceDegree (b) Classe intensi cationMode

Figure C.4: Instances des classes du module lfrep

(a) Classe realizationDegree (b) Classe spatialSpeci cation

Figure C.5: Instances des classes du module lfrep

module lfrep (lexical functions representation), qui est l'implémentation des propriétés des FL (Section 8.2) : inspiré par la TST (Chapitre 3) et par la

(a) ClassespecializationType (b) Classestandardness

Figure C.6: Instances des classes du module

(a) ClassesyntacticActant (b) ClassesyntacticActantModi ers

Figure C.7: Instances des classes du module

théorie de FL (Chapitre 4) ;

module lfam (lexical function family), créé pour faire l'organisation des FL en classes syntaxiques/fonctionnelles similaires (Section 8.3) : basé sur la division du RLF en familles de FL ;

module lfsem (lexical function semantic perspective), pour l'encodage des perspectives sémantiques des FL (Section 8.4) : implémentation des perspectives sémantiques des FL présentées à la section 4.7 ;

module lfrel (lexical function relations), pour la représentation des relations entre unités lexicales (Section 8.5) : inspiré par la nécessité de la représentation des relations lexicales paradigmatiques et syntagmatiques présentées à la section 1.8. L'ontologie demon (Sous-section 6.5.2) a été utilisée pour la représentation conceptuelle des formes et des sens des mots.

Ces quatre modules ont été utilisés pour la transformation du RLF dans un format compatible avec les formalismes du Web sémantique, tel qu'expliqué à la section suivante.

1.3. Le processus de transformation du Réseau lexical du français

Nous avons reçu les données du RLF sous la forme de tableaux de bases de données relationnelles. Les principaux tableaux sont les suivants :

lf (les informations sur les FL) : nom, standardness, famille des FL ;

senses_lf : les applications des FL à des sens spécifiques ;

senses_lf_targets : chacune des relations dans senses_lf appliquée à des sens cibles pour former des relations paradigmatiques et syntagmatiques ;

senses : l'identificateur de chaque sens connecté à son vocable ;

vocables : la liste des vocables présents dans le RLF.

Un programme Java a été créé pour nettoyer les chiers des informations non linguistiques, comme le nom de l'utilisateur qui a fait la dernier changement, la date de changement, etc. Des chiers en format JSON[2] (JavaScript Object Notation) ont été créés.

À partir des chiers en format JSON, trois ontologies (c'est-à-dire, trois chiers en format RDF/OWL ayant la terminaison .owl) ont été créées en utilisant un programme JAVA :

lfrlf-vocables-senses.owl : ontologie qui contient les informations de vocables et de sens extraites du RLF. Un exemple du contenu de cette ontologie est montré par la gure 8.10, au chapitre 8 ;

lfrlf.owl : informations sur les FL extraites du RLF. Après la création de cette ontologie, le programme Protégé a été utilisé pour l'encodage manuel des informations suivantes pour chaque FL simple (environ 100 FL) : la perspective sémantique, la famille, la dimension (paradigmatique ou syntagmatique) et des informations particulières à chaque type de FL, comme l'actant syntaxique, la dimension de l'intensi cation (pour les FL Magn), etc. Après la représentation des FL simples, des composantes ont été créées pour connecter les FL complexes (environ 500) à leurs FL simples constituantes, tel que expliqué à la sous-section 8.2.1 ;

lfsr.owl : informations sur les relations lexicales paradigmatiques et syntagmatiques. Les informations présentes dans les cinq chiers JSON présentés ci-dessus ont été combinées pour créer cette ontologie. Un exemple du contenu de cette ontologie est présenté à la section 8.5.

2. http ://www.json.org/

RÉFÉRENCES

Ackrill, J. (1975). Aristotle : Categories and De Interpretatione. Clarendon Aristotle Series. Clarendon Press. 170 p.

Alonso Ramos, M. (2006). Towards a dynamic way of learning collocations in a second language. Dans C. O. Elisa Corino, Carla Marello (dir.), Proceedings of the 12th EURALEX International Congress, 909 921., Torino, Italy. Edizioni dell'Orso.

Alonso Ramos, M., O., R. et L., W. (2008). Using semantically annotated corpora to build collocation resources. Dans Proceedings of LREC, 1154 1154.

Antoniou, G. et van Harmelen, F. (2004). A Semantic Web Primer. Cambridge, MA, USA : MIT Press. 238 p.

Apresjan, J., Boguslavsky, I., Iomdin, L. et Tsinman, L. (2000). Lexical functions in NLP : Possible uses 1http ://cl.iitp.ru/bibitems/LF_uses.pdf , 1 13.

Baader, F., Horrocks, I. et Sattler, U. (2009). Description logics. Dans Staab et Studer (2009), 21 43.

Baker, C. F., Fillmore, C. J. et Cronin, B. (2003). The Structure of the FrameNet Database. International Journal of Lexicography, 16(3), 281 296.

Baker, C. F., Fillmore, C. J. et Lowe, J. B. (1998). The Berkeley FrameNet project. Dans Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational

Linguistics - Volume 1, ACL '98, 86 90., Stroudsburg, PA, USA. Association for Computational Linguistics.

Bentivogli, L. et Pianta, E. (2004). Extending WordNet with syntagmatic information. Dans Second Global WordNet Conference, 47 53.

Boguslavsky, I., Iomdin, L. et Sizov, V. (2004). Multilinguality in ETAP-3 : Reuse of lexical resources. Dans Proceedings of the Workshop on Multilingual Linguistic Ressources, MLR '04, 7 14., Stroudsburg, PA, USA. Association for Computational Linguistics.

Bohnet, B. et Wanner, L. (2010). Open source graph transducer interpreter and grammar development environment. Dans Proceeding of the LREC-2010, 211 218.

Bolshakov, I. et Gelbukh, A. (1998). Lexical functions in Spanish. Dans Proceedings of CIC-98, Simposium Internacional de Computación, 383 395.

Bordes, A., Weston, J., Collobert, R. et Bengio, Y. (2011). Learning structured embeddings of knowledge bases. Dans Proceedings of the Twenty-Fifth AAAI Conference on Arti cial Intelligence, 301 306.

Borges, J. (1981). Funes el memorioso. Dos Amigos. 20 p.

Brentano, F. C. (1978). Aristotle and His World View. University of California Press. Edited and translated by R. George and R. M. Chisholm. 138 p.

Buchholz, S. et Marsi, E. (2006). Conll-x shared task on multilingual dependency parsing. Dans Proceedings of the Tenth Conference on Computational Natural Language Learning, CoNLL-X '06, 149 164., Stroudsburg, PA, USA. Association for Computational Linguistics.

Candito, M., Nivre, J., Denis, P. et Anguiano, E. H. (2010). Benchmarking of Statistical Dependency Parsers for French. Dans *Proceedings of the 23rd International Conference on Computational Linguistics : Posters*, COLING '10, 108 116., Stroudsburg, PA, USA. Association for Computational Linguistics.

Chomsky, N. (1957). *Syntactic Structures* The Hague : Mouton. 15 p.

Chomsky, N. (2002). Perspectives on language and mind. *J. Mach. Learn. Res.*, 9, 1871 1874.

Choueka, Y. (1988). Looking for needles in a haystack or locating interesting collocational expressions in large textual databases. Dans C. Fluhr et D. E. Walker (dir.). *RIAO*, 609 624. CID.

Church, K. W. et Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1), 22 29.

Cibelli, E., Xu, Y., Austerweil, J. L., Gri ths, T. L. et Regier, T. (2016). The Sapir-Whorf hypothesis and probabilistic inference : Evidence from the domain of color. *PLoS ONE*, 11(8), 1 28.

Cimiano, P., Buitelaar, P., McCrae, J. et Sintek, M. (2011). Lexinfo : A declarative model for the lexicon-ontology interface. *Web Semantics : Science, Services and Agents on the World Wide Web*, 9(1), 29 51.

Cimiano, P., Haase, P., Herold, M., Mantel, M. et Buitelaar, P. (2007). LexOnto : A Model for Ontology Lexicons for Ontology-based NLP. Dans *Proceedings of OntoLex - From Text to Knowledge : The Lexicon/Ontology Interface (Workshop at the International Semantic Web Conference)*, 1 12.

Coseriu, E. (1986). *Introducción a la lingüística*. Biblioteca románica hispánica : Manuales. Gredos. 178 p.

Crabbé, B., Duchier, D., Gardent, C., Le Roux, J. et Parmentier, Y. (2013). XMG : eXtensible MetaGrammar. Computational Linguistics, 39(3), 591 629.

Crouch, D. et King, T. H. (2006). Semantics Via F-Structure Rewriting. Dans M. Butt et T. H. King (dir.). Lexical Functional Grammar Conference 2006, 145 165.

Dalrymple, M., Lamping, J. et Saraswat, V. (1993). LFG semantics via constraints. Dans Sixth Conference of the European Chapter of the Association for Computational Linguistics, 97 105.

Davis, R., Shrobe, H. E. et Szolovits, P. (1993). What is a knowledge representation ? AI Magazine, 14(1), 17 33.

Delli Bovi, C., Collados, J. C., Raganato, A. et Navigli, R. (2017). Eurosense : Automatic harvesting of multilingual sense annotations from parallel text. Dans Proceedings of 55th annual meeting of the Association for Computational Linguistics (ACL 2017), 594 600., Vancouver, Canada.

Dendien, J. et Pierrel, J.-M. (2003). Le trésor de la langue française informatisé : un exemple d'informatisation d'un dictionnaire de langue de référence. Traitement Automatique des Langues (T.a.l.), 44, 11 37.

Eco, U. (1999). Kant et l'ornithorynque. Essais Etranger. Grasset. 673 p.

Elman, J. L., Bates, E. A., Johnson, M. H., Karmilo -Smith, A., Parisi, D. et Plunkett, K. (1996). Rethinking Innateness : A Connectionist Perspective on Development. MIT Press. 447 p.

Eluerd, R. (2000). La lexicologie (Que sais-je ?). Paris : Presses Universitaires de France (PUF). 693 p.

Erbach, G. et Krenn, B. (1993). Idioms and support verb constructions in HPSG. Computerlinguistik an der Universität des Saarlandes. Univ. des Saarlandes, Computerlinguistik. 24 p.

Evans, V., Bergen, B. K. et Zinken, J. (2007). The cognitive linguistics enterprise : an overview. In V. Evans, B. K. Bergen, et J. Zinken (dir.), The cognitive linguistics reader (1 35.

Fauconnier, G. et Turner, M. (2002). The Way We Think : Conceptual Blending and the Mind's Hidden Complexities. Basic Books. 444 p.

Fellbaum, C. (dir.) (1998). WordNet An Electronic Lexical Database. Cambridge, MA ; London : The MIT Press. 423 p.

Fillmore, C. J. (1977). Scenes-and-frames semantics. Numéro 59 de Fundamental Studies in Computer Science. North Holland Publishing.

Fodor, J. (1975). The Language of Thought. Language and thought series. Harvard University Press. 224 p.

Fodor, J. D., Fodor, J. A. et Garrett, M. F. (1975). The psychological unreality of semantic representations. Linguistic Inquiry , 6, 515 531.

Fonseca, A. et Sadat, F. (2014). A comparative study of di erent classi cation methods for the identi cation of brazilian portuguese multiword expressions. Dans Proceedings of the First Workshop on Computational Approaches to Compound Analysis (ComAComA 2014) (53 62., Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Fonseca, A., Sadat, F. et Blondin Massé, A. (2014). Identifying portuguese multiword expressions using di erent classi cation algorithms - a comparative analysis. Dans Proceedings of the 4th International Workshop on Computational

Terminology (Computerm), 104 113., Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Fonseca, A., Sadat, F. et Lareau, F. (2016a). Lexfom : a lexical functions ontology model. Dans Proceedings of the Fifth Workshop on Cognitive Aspects of the Lexicon (CogALex), COLING, 145 155., Osaka, Japan.

Fonseca, A., Sadat, F. et Lareau, F. (2016b). A lexical ontology to represent lexical functions. Dans Proceedings of the 2nd Workshop on Language and Ontologies (OntoLex), LREC, 69 73., Portoroº, Slovenia.

Fonseca, A., Sadat, F. et Lareau, F. (2017). Combining dependency parsing and a lexical network based on lexical functions for collocations identi cation, 447 461. Lecture Notes in Arti cial Inteligence. Proceedings of the EUROPHRAS-2017. Springer International Publishing : London, UK.

Fonseca, A., Sadat, F. et Lareau, F. (2018). Retrieving Information from the French Lexical Network in RDF/OWL format. Dans 11th International Conference on Language Resources and Evaluation (LREC-2018), 4408 4412., Miyazaki, Japan.

Fontenelle, T. (2012). WordNet, FrameNet and Other Semantic Networks in the International Journal of Lexicography - The Net Result ? International Journal of Lexicography 25, 437 449.

Francopoulo, G., George, M., Calzolari, N., Monachini, M., Bel, N., Pet, Y. et Soria, C. (2006). Lexical Markup Framework (LMF). Dans Proceedings of the International Conference on Language Resources and Evaluation - LREC-2006, 233 236.

Frege, G. (1948). Sense and reference. Philosophical Review, 57(3), 209 230.

Fuchs, C. et Le Go c, P. (1992). Les linguistiques contemporaines : repères théoriques. Langue linguistique communication. Paris : Hachette Supérieur, cop. 1992 (25-Baume-les-Dames). 160 p.

Gangemi, A., Navigli, R. et Velardi, P. (2003). The ontowordnet project : Extension and axiomatization of conceptual relations in wordnet. Dans On The Move to Meaningful Internet Systems 2003 : CoopIS, DOA, and ODBASE - OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003, 820 838.

Garcia, M., García-Salido, M. et Alonso-Ramos, M. (2017). Using bilingual word-embeddings for multilingual collocation extraction. Dans Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017), 21 30., Valencia, Spain. Association for Computational Linguistics.

Genesereth, M. R., Fikes, R. E. et al. (1992). Knowledge interchange format-version 3.0 : reference manual. 1 68.

Grali«ski, F., Savary, A., Czerepowicka, M. et Makowiecki, F. (2010). Computational lexicography of multi-word units : How e cient can it be ? Dans Proceedings of the Workshop on Multiword Expressions : from Theory to Applications (MWE 2010), 1 9., Beijing, China. Association for Computational Linguistics.

Grégoire, N. (2010). Duelme : a dutch electronic lexicon of multiword expressions. Language Resources and Evaluation, 44(1), 23 39.

Grossmann, F. (2011). Didactique du lexique : état des lieux et nouvelles orientations. Pratiques (Didactique du français), 149/150(2), 163 183.

Gruber, T. R. (1992). Ontolingua : A mechanism to support portable ontologies. Stanford University, Knowledge Systems Laboratory Stanford. Rapport technique. 61 p.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing. Int. J. Hum.-Comput. Stud., 43(5-6), 907 928.

Harnad, S. (1990). The symbol grounding problem. Physica D, 42, 335 346.

Harnad, S. (2005). To cognize is to categorize : Cognition is categorization. UQAM Summer Institute in Cognitive Sciences on Categorization. 30 June - 11 July 2003 Event Dates : 30 June - 11 July 2003.

Harnad, S. (2010). Eliminating the concept concept. Behavioral and Brain Sciences, 33(2-3), 213 214.

Harnad, S. (2012). Alan turing and the "hard" and "easy" problem of cognition : Doing and feeling. CoRR, abs/1206.3658

Hausmann, F. J. (1979). Un dictionnaire des collocations est-il possible ? Travaux de linguistique et de littérature, 17(1), 187 195.

Heid, U. et Raab, S. (1989). Collocations in multilingual generation. Dans Proceedings of the Fourth Conference of ACL, European Chapter, Manchester, 10-12 April 1989, Manchester.

Hendler, J. et van Harmelen, F. (2008). The semantic web : Webizing knowledge representation. Dans van Harmelen et al. (2008), 821 839.

Heylen, D., Maxwell, K. G. et Verhagen, M. (1994). Lexical functions and machine translation. Dans Proceedings of the 15th Conference on Computational Linguistics - Volume 2, COLING '94, 1240 1244., Stroudsburg, PA, USA. Association for Computational Linguistics.

Hirst, G. (2004). Ontology and the Lexicon, 209 229. Springer Berlin Heidelberg : Berlin, Heidelberg

Jackendo , R. (1997). Twistin' the night away. Language, 73(3), 534 559.

Johansson, R. et Piña, L. N. (2015). Embedding a semantic network in a word space. Dans *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, 1428–1433.

Johnson, M. (1990). *The Body in the Mind : The Bodily Basis of Meaning, Imagination, and Reason*. Philosophy, psichology, cognitive sciencies. University of Chicago Press. 233 p.

Jousse, A.-L. (2010). *Modèle de structuration des relations lexicales fondé sur le formalisme des fonctions lexicales* (Thèse de doctorat). Thèse de doctorat dirigée par Sylvain Kahane et Alain Polguere, Université de Montréal et Université Paris Diderot (Paris 7). 279 p.

Kaplan, R. et Bresnan, J. (1982). Lexical-functional grammar : A formal system for grammatical representation. In J. Bresnan (dir.), *The Mental Representation of Grammatical Relations*, 173–281. Cambridge, MA : MIT Press.

Kavalec, M. et Svátek, V. (2005). A study on automated relation labelling in ontology learning. In P. Buitelaar, P. Cimiano, et B. Magnini (dir.), *Ontology Learning from Text : Methods, Evaluation and Applications*, numéro 123 de Frontiers in Arti cial Intelligence and Applications, 44–58. IOS Press.

Kiefer, F. (1988). Linguistic, conceptual and encyclopedic knowledge : Some implications for lexicography. Dans T. Magay et J. Zigány (dir.), *Proceedings of the 3rd EURALEX International Congress*, 1–10., Budapest, Hungary. Akadémiai Kiadó.

Kifer, M. et Lausen, G. (1989). F-logic : A higher-order language for reasoning about objects, inheritance, and scheme. Dans *Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data*, 134–146.

Kleene, S. C. (1952). Introduction to Metamathematics. North Holland. 560 p.

Kolesnikova, O. (2011). Automatic Extraction of Lexical Functions. (Thèse de doctorat). Thèse de doctorat dirigée par Alexander Gelbukh, Instituto Politecnico Nacional - Centro de Investigacion en Computacion, Mexico. 116 p.

Lako , G. (1987). Cognitive models and prototype theory, 63 100. Cambridge University Press : Cambridge.

Lako , G. (1991). Cognitive versus generative linguistics : How commitments in uence results. Language and communication, 11(1/2), 53 62.

Lako , G. (1993). The Contemporary Theory of Metaphor, 202 251. Cambridge University Press : Cambridge.

Lambrey, F. (2016). Implémentation des collocations pour la réalisation de texte multilingue. (Thèse de doctorat). Mémoire de maîtrise dirigée par François Lareau, Université de Montréal. 211 p.

Lambrey, F. et Lareau, F. (2015). Le traitement des collocations en génération de texte multilingue. 579 585.

Lareau, F., Dras, M., Börschinger, B. et Dale, R. (2011). Collocations in multilingual natural language generation : Lexical functions meet lexical functional grammar. Dans Proceedings of ALTA'11, 95 104., Canberra.

Lareau, F., Dras, M., Börschinger, B. et Turpin, M. (2012). Implementing lexical functions in XLE. Dans M. Butt et T. H. King (dir.). Proceedings of LFG12, 362 382., Denpasar, Indonesia.

Lareau, F. et Wanner, L. (2007). Towards a generic multilingual dependency grammar for text generation. Dans T. H. King et E. M. Bender (dir.). Proceedings of the GEAF07 Workshop, 203 223., Stanford. CSLI.

Lefrançois, M. et Gandon, F. (2012). Ilexicon : toward an ecd-compliant interlingual lexical ontology described with semantic web formalisms. CoRR, abs/1204.5316

Lehmann, A. et Martin-Berthet, F. (1998). Introduction à la lexicologie : sémantique et morphologie. Collection Lettres supérieures. Dunod. 201 p.

Lemmens, M. (2015). Cognitive semantics. In N. Riemer (dir.), Routledge Handbook of Semantics, 90 105. Routledge. version auteur, avant publication

Lenat, D. B. et Guha, R. V. (1991). The evolution of cycl, the cyc representation language. SIGART Bull., 2(3), 84 87.

L'Homme, M. C. (2008). Le dicoinfo : méthodologie pour une nouvelle génération de dictionnaires spécialisés. Traduire, 217, 78 103.

Lichte, T., Parmentier, Y., Petitjean, S., Savary, A. et Waszczuk, J. (2016). Separating the regular from the idiosyncratic : A constraint-based lexical encoding of MWEs using XMG. Dans PARSEME 6th general meeting, 7-8 April 2016, Struga, FYR Macedonia, 1 4.

Lifschitz, V., Morgenstern, L. et Plaisted, D. A. (2008). Knowledge representation and classical logic. Dans van Harmelen et al. (2008), 3 88.

Liu, H. et Singh, P. (2004). Conceptnet &mdash ; a practical commonsense reasoning tool-kit. BT Technology Journal, 22(4), 211 226.

Lucy, J. A. (1998). Sapir-Whorf hypothesis. Routledge Encyclopedia of Philosophy, 471 485.

Lux-Pogodalla, V. et Polguère, A. (2011). Construction of a French Lexical Network : Methodological Issues. Dans First International Workshop on Lexical Resources, WoLeR 2011, 54 61., Ljubljana, Slovenia.

Lyons, J. (1977). Semantics 1 Cambridge, United Kingdom : Cambridge University Press. 371 p.

Lyons, J. (1995). Linguistic Semantics : An Introduction. Cambridge University Press. 376 p.

L'Homme, M.-C. et Lanneville, M. (2014). DicoEnviro. dictionnaire fondamental de l'environnement. Consulté à l'adresse http ://olst. ling. umontreal. ca/cgi-bin/dicoenviro/search. cgi.

MacGregor, R. M. et Bates, R. (1987).The Loom knowledge representation language. Rapport technique, Information Science Institute, University of Southern California. 21 p.

Manning, C. D. et Schütze, H. (1999).Foundations of Statistical Natural Language Processing Cambridge, MA, USA : MIT Press. 680 p.

Marchetti, A., Ronzano, F., Tesconi, M. et Minutoli, S. (2008).Formalizing Knowledge by Ontologies : OWL and KIF Rapport technique, KYOTO Working paper : nr 003, WP2 System design, Technical Rapport

Margolis, E. et Lawrence, S. (2005). Concepts.Stanford Encyclopedia of Philosophy. [Publié en ligne ; consulté le 14-Juin-2013].

Marie Candito, E. H. A. et Seddah, D. (2011). A word clustering approach to domain adaptation : E ective parsing of biomedical texts. Dans Proceedings of the 12th International Conference on Parsing Technologies, 37 42., Vancouver, Canada.

Maziarz, M., Szpakowicz, S. et Piasecki, M. (2012). Semantic relations among adjectives in polish wordnet 2.0 : a new relation set, discussion and evaluation. Cogntive Studies, 12, 149 179.

McCrae, J., Spohr, D. et Cimiano, P. (2011). Linking lexical resources and ontologies on the semantic web with lemon. Dans Proceedings of the 8th Extended Semantic Web Conference on The Semantic Web : Research and Applications - Volume Part I, ESWC'11, 245 259., Berlin, Heidelberg. Springer-Verlag.

Mcdonald, S. et Ramscar, M. (2001). Testing the Distributional Hypothesis : The In uence of Context on Judgements of Semantic Similarity. Dans In Proceedings of the 23rd Annual Conference of the Cognitive Science Society, 611 6.

Mel'£uk, I. (1992). Paraphrase et lexique : la théorie Sens-Texte et le Dictionnaire explicatif et combinatoire, 9 58. Presses de l'Université de Montréal : Montreal, Canada.

Mel'£uk, I. (1995). Lexical Functions : A Tool for the description of Lexical Relations in the Lexicon. In L. Wanner (dir.), Lexical Functions in Lexicography and Natural Language Processing, 37 102. Amsterdam/Philadelphia : John Benjamins.

Mel'£uk, I. (1997). Vers une linguistique sens-texte. Collège de France Paris : Leçon inaugurale. Collège de France. 43 p.

Mel'£uk, I. (2015). Clichés, an understudied subclass. De Gruyter Mouton, 6, 55 86.

Mel'£uk, I., Clas, A. et Polguère, A. (1995). Introduction à la lexicologie explicative et combinatoire. Louvain-la-Neuve (Belgique) : Duculot. 256 p.

Mel'£uk, I. (1998). Collocations and lexical functions. Phraseology. Theory, Analysis and Applications, 23 53.

Mikolov, T., Chen, K., Corrado, G. et Dean, J. (2013). E cient estimation of word representations in vector space. CoRR, abs/1301.3781

Mili¢evi¢, J. (2007). La paraphrase : modélisation de la paraphrase langagière Sciences pour la communication. Lang. 400 p.

Moon, R. (1998). Fixed Expressions and Idioms in English : A Corpus-based Approach. Oxford studies lexicography and lexicology. Clarendon Press. 338 p.

Moreno, P., Ferraro, G. et Wanner, L. (2013). Can we determine the semantics of collocations without using semantics ? Dans Proceedings of The 3rd Biennial Conference On Electronic Lexicography, eLex 2013, 306 121.

Moro, A. et Navigli, R. (2015). Semeval-2015 task 13 : Multilingual all-words sense disambiguation and entity linking. Dans D. M. Cer, D. Jurgens, P. Nakov, et T. Zesch (dir.). Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015, 288 297. The Association for Computer Linguistics.

Murphy, M. (2003). Semantic Relations and the Lexicon : Antonymy, Synonymy and other Paradigms. Cambridge University Press. 293 p.

Navigli, R. (2009). Word sense disambiguation : A survey. ACM Comput. Surv., 41(2), 10 :1 10 :69.

Navigli, R. et Ponzetto, S. P. (2012). Babelnet : The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Arti cial Intelligence , 193, 217 250.

Nica, I., Martí, M. A., Montoyo, A. et Vázquez, S. (2004). Enriching ewn with syntagmatic information by means of wsd. Dans LREC, 153 156. European Language Resources Association.

Odijk, J. (2004). A proposed standard for the lexical representation of idioms. Dans G. Williams et S. Vessier (dir.). Proceedings of the 11th EURALEX In-

ternational Congress, 153 164., Lorient, France. Universite de Bretagne-Sud, Faculte des lettres et des sciences humaines.

Orliac, B. (2006). Colex. un outil d'extraction de collocations spécialisées basé sur les fonctions lexicales. Terminology, 12, 261 280.

Padó, S. (2007). Cross-Lingual Annotation Projection Models for Role-Semantic Information, volume 21 de Saarbrücken Dissertations in Computational Linguistics and Language Technology. German Research Center for Arti cial Intelligence and Saarland University. 213 p.

Partee, B. (1976). Montague grammar. Academic Press. 386 p.

Pease, A., Niles, I. et Li, J. (2002). The suggested upper merged ontology : A large ontology for the semantic web and its applications. Dans In Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web, 10 4.

Picoche, J. (1977). Precis de lexicologie francaise. Nathan [Paris]. 181 p.

Polguère, A. (2000). Towards a theoretically-motivated general public dictionary of semantic derivations and collocations for french. Dans E. L. C. R. Ulrich Heid, Stefan Evert (dir.). Proceedings of the 9th EURALEX International Congress, s 517 527., Stuttgart, Germany. Institut für Maschinelle Sprachverarbeitung.

Polguère, A. (1998). La théorie sens-texte. Dialangue, 8-9(1), 9 30.

Polguère, A. (2003). Étiquetage sémantique des lexies dans la base de données dico. Traitement Automatique des Langues, 44, 39 68.

Polguère, A. (2011). Perspective épistémologique sur l'approche linguistique Sens-Texte, 79 114. Peeters Publishers : Leuven, Belgium.

Polguère, A. (2014). From writing dictionaries to weaving lexical networks. International Journal of Lexicography, 27(4), 396 418.

Pollard, C. et Sag, I. A. (1994). Head-Driven Phrase Structure Grammar Chicago : University of Chicago Press. 454 p.

Pottier, B. (1992). Sémantique générale Linguistique Nouvelle. Presses Universitaires de France. 237 p.

Proost, K. (2007). Conceptual Structure in Lexical Items : The Lexicalisation of Communication Concepts in English, German, and Dutch Pragmatics & beyond. J. Benjamins Publishing Company. 304 p.

Pylyshyn, Z. (1986). Computation and Cognition : Toward a Foundation for Cognitive Science Bradford Books. MA : The MIT Press. 292 p.

Quillian, R. (1968). Semantic memory Semantic Information Processing 227 270.

Ramisch, C. (2012). A generic and open framework for multiword expressions treatment : from acquisition to applications. (Un environnement générique et ouvert pour le traitement des expressions polylexicales) (Thèse de doctorat). Grenoble Alpes University, France. 234 p.

Rescorla, M. (2015). The computational theory of mind. Stanford Encyclopedia of Philosophy [Publié en ligne ; consulté le 07-May-2017].

Reus, B. (2016). The Church-Turing Thesis, 123 148. Springer Publishing Company, Incorporated, (1st éd.).

Richardson, S. D., Dolan, W. B. et Vanderwende, L. (1998). Mindnet : Acquiring and structuring semantic information from text. Dans Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2, 1098 1102., Montreal, Quebec, Canada. Association for Computational Linguistics.

Robin, L., Moreau, M. et Chatelet, F. (1977). *Apologie de Socrate : Criton ; Phédon*. Collection Idées : Philosophie. Gallimard. 256 p.

Rosch, E. (1975). Cognitive representations of semantic categories. *Journal of Experimental Psychology : General*, *104*, 192–233.

Ruppenhofer, J., Ellsworth, M., Petruck, M. R., Johnson, C. R. et Scheffczyk, J. (2006). *FrameNet II : Extended Theory and Practice*. Berkeley, California : International Computer Science Institute. Distributed with the FrameNet data. 219 p.

Sahlgren, M. (2006). *The Word-Space Model : Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations Between Words in High-dimensional Vector Spaces*. SICS dissertation series. Department of Linguistics, Stockholm University. 146 p.

Saussure, F. d., Riedlinger, A., Sechehaye, A., Bally, C. et De Mauro, T. (1972). Cours de linguistique générale. Payothèque. Reproduction en fac-similé de la 3e édition, 1931. L'édition de 1972 est augmentée par Tullio De Mauro d'une introduction, de notes et commentaires, traduits de l'italien, et d'une bibliogr. pp. 481-495. Index.

Schwab, D., Tze, L. L. et Lafourcade, M. (2007). Les vecteurs conceptuels, un outil complémentaire aux réseaux lexicaux. Dans *Proceedings of TALN'07 : Traitement Automatique des Langues Naturelles*, 293–302. ATALA.

Searle, L., Keeler, M., Sowa, J., Deluagch, H. et Lukose, D. (1997). *Fulfilling Peirce's dream : Conceptual structures and communities of inquiry*, 1–11. Springer Berlin Heidelberg : Berlin, Heidelberg

Smadja, F., McKeown, K. R. et Hatzivassiloglou, V. (1996). Translating colloca-

tions for bilingual lexicons : A statistical approach. *Comput. Linguist.*, *22*(1), 1–38.

Sowa, J. F. (1984). *Conceptual Structures : Information Processing in Mind and Machine*. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc. 481 p.

Sowa, J. F. (1997). *Peircean foundations for a theory of context*, 41–64. Springer Berlin Heidelberg : Berlin, Heidelberg.

Sowa, J. F. (2008). Conceptual graphs. Dans van Harmelen *et al.* (2008), 213–237.

Sowa, J. F. (2010). *The Role of Logic and Ontology in Language and Reasoning*, 231–263. Springer Netherlands : Dordrecht

Speer, R. et Havasi, C. (2013). *ConceptNet 5 : A Large Semantic Network for Relational Knowledge*, 161–176. Springer Berlin Heidelberg : Berlin, Heidelberg

Staab, S. et Studer, R. (2009). *Handbook on Ontologies* (2nd éd.). Springer Publishing Company, Incorporated. 811 p.

Tesnière, L. (1959). *Eléments de Syntaxe Structurale*. Paris : Klincksieck. 670 p.

Tomokiyo, M., Weyer-Brown, P. et Mangeot, M. (2006). Représentation sémantique de lexique pour un dictionnaire de traduction manuelle et automatique. Dans *Actes des Aspects méthodologiques pour l'élaboration de lexiques unilingues et multilingues*, 1–12., Bertinoro, Italie.

Tremblay, O. (2009). *Une ontologie des savoirs lexicologiques pour l'élaboration d'un module de cours en didactique du lexique*. (Thèse de doctorat). Université du Québec à Montréal. 693 p.

Tremblay, O. et Polguère, A. (2014). Une ontologie linguistique au service de la didactique du lexique. Dans L. H. G. G. J. M. e. S. P. F. Neveu, P. Blumenthal (dir.). *4e Congrès Mondial de Linguistique Française (CMLF 2014) Berlin, Allemagne, July 2014*, 1173–1188. EDP Sciences.

Turing, A. M. (1936). On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, *2*(42), 230–265.

Turing, A. M. (1995). : chapitre Computing Machinery and Intelligence, 11–35. Cambridge, MA, USA : MIT Press

Urieli, A. (2013). *Robust French syntax analysis : reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. (Thèse de doctorat). Université de Toulouse. 248 p.

Vachek, J. (1966). *The linguistic school of Prague : an introduction to its theory and practice*. Indiana University studies in the history and theory of linguistics. Indiana University Press. 184 p.

Valente, R. S. (2002). *La "lexicologie explicative et combinatoire" dans le traitement des unités lexicales spécialisées. Thèse de Doctorat, Département de linguistique et traduction*. Université de Montréal. 322 p.

van Harmelen, F., Lifschitz, V. et Porter, B. W. (dir.) (2008). *Handbook of Knowledge Representation*, volume 3 de *Foundations of Artificial Intelligence*. Elsevier. 1034 p.

Vandaele, S. et Lubin, L. (2005). Approche cognitive de la traduction dans les langues de spécialité : vers une systématisation de la description de la conceptualisation métaphorique. *Meta : journal des traducteurs / Meta : Translators' Journal*, *50*, 415–431.

Villavicencio, A., Copestake, A., Waldron, B. et Lambeau, F. (2004). The lexical encoding of mwes. Dans T. Tanaka, A. Villavicencio, F. Bond, et A. Korhonen (dir.). *Proceedings of the ACL 2004 Workshop on Multiword Expressions : Integrating Processing*, 80–87. ACL.

Vincze, V., Almási, A. et Szauter, D. (2008). Comparing wordnet relations to lexical functions. Dans *Proceedings of The Fourth Global WordNet Conference*, 462–473.

Žolkovskij, A. et Mel'čuk, I. (1965). O vozmozhnom metode i instrumentax semanticheskogo sinteza [on a possible method an instruments for semantic synthesis (of texts)]. *Scientific and Technological Information*, *6*, 23–28.

Žolkovskij, A. et Mel'čuk, I. (1967). O semanticheskogo sinteza [on semantic synthesis (of texts)]. *Problemy kybernetiki [Problems of Cybernetics]*, *19*, 177–238.

Wanner, L. (2004). Towards automatic fine-grained semantic classification of verb-noun collocations. *Natural Language Engineering*, *10*(2), 95–143.

Wanner, L. et Bateman, J. A. (1990). A collocational based approach to salience-sensitive lexical selection. Dans *Workshop On Natural Language Generation*, 31–38.

Wanner, L., Bohnet, B., Bouayad-Agha, N., Lareau, F. et Nicklaß, D. (2010). MARQUIS : Generation of user-tailored multilingual air quality bulletins. *Applied Artificial Intelligence*, *24*(10), 914–952.

Wanner, L., Bohnet, B. et Giereth, M. (2006). What is beyond collocations ? insights from machine learning experiments. Dans C. O. Elisa Corino, Carla Marello (dir.). *Proceedings of the 12th EURALEX International Congress*, 1071–1087., Torino, Italy. Edizioni dell'Orso.

Wierzbicka, A. (1996). *Semantics : Primes and Universals : Primes and Universals*. Oxford University Press, UK. 512 p.

Wittgenstein, L. (1953). *Philosophical Investigations. (Translated by Anscombe, G.E.M.)*. Oxford : Basil Blackwell. 250 p.