Université de Montréal

Création automatique d'un dictionnaire des régimes des verbes du français

Par
Naïma Hassert

Département de linguistique et de traduction, Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de maîtrise ès arts (M.A.) en linguistique

Juin 2023

Université de Montréal
Département de linguistique et de traduction, Faculté des arts et des sciences

---

Ce mémoire intitulé

Création automatique d'un dictionnaire des régimes des verbes du français

Présenté par

Naïma Hassert

A été évalué par un jury composé des personnes suivantes

Patrick Drouin
Président-rapporteur

François Lareau
Directeur de recherche

Dominic Forest
Membre du jury

Résumé

Les dictionnaires de valence sont utiles dans plusieurs tâches en traitement automatique des langues. Or, les dictionnaires de qualité de ce type sont créés au moins en partie manuellement ; ils nécessitent donc beaucoup de ressources et sont difficiles à mettre à jour. De plus, plusieurs de ces ressources ne prennent pas en compte les différents sens des lemmes, qui sont pourtant importants puisque les arguments sélectionnés ont tendance à varier selon le sens du verbe. Dans ce mémoire, nous créons automatiquement un dictionnaire de valence des verbes du français qui tient compte de la polysémie. Nous extrayons 20 000 exemples de phrases pour chacun des 2 000 verbes les plus fréquents du français. Nous obtenons ensuite les plongements lexicaux de ces verbes en contexte à l'aide d'un modèle de langue monolingue et de deux modèles de langue multilingues. Puis, nous utilisons des algorithmes de regroupement pour induire les différents sens de ces verbes. Enfin, nous analysons automatiquement les phrases à l'aide de différents analyseurs syntaxiques afin de trouver leurs arguments. Nous déterminons que la combinaison du modèle de langue français *CamemBERT* et d'un algorithme de regroupement agglomératif offre les meilleurs résultats dans la tâche d'induction de sens (58,19 % de F_1 B^3), et que pour l'analyse syntaxique, *Stanza* est l'outil qui a les meilleures performances (83,29 % de F_1). En filtrant les cadres syntaxiques obtenus à l'aide d'une estimation de la vraisemblance maximale, une méthode statistique très simple qui permet de trouver les paramètres les plus vraisemblables d'un modèle de probabilité qui explique nos données, nous construisons un dictionnaire de valence qui se passe presque complètement d'intervention humaine. Notre procédé est ici utilisé pour le français, mais peut être utilisé pour n'importe quelle autre langue pour laquelle il existe suffisamment de données écrites.

Mots-clés: induction de sens, valence, lexicographie computationnelle

Abstract

Valency dictionaries are useful for many tasks in automatic language processing. However, quality dictionaries of this type are created at least in part manually; they are therefore resource-intensive and difficult to update. In addition, many of these resources do not take into account the different meanings of lemmas, which are important because the arguments selected tend to vary according to the meaning of the verb. In this thesis, we automatically create a French verb valency dictionary that takes polysemy into account. We extract 20 000 example sentences for each of the 2 000 most frequent French verbs. We then obtain the lexical embeddings of these verbs in context using a monolingual and two multilingual language models. Then, we use clustering algorithms to induce the different meanings of these verbs. Finally, we automatically parse the sentences using different parsers to find their arguments. We determine that the combination of the French language model *CamemBERT* and an agglomerative clustering algorithm offers the best results in the sense induction task (58.19 % of F_1 B^3), and that for syntactic parsing, *Stanza* is the tool with the best performance (83.29 % of F_1). By filtering the syntactic frames obtained using maximum likelihood estimation, a very simple statistical method for finding the most likely parameters of a probability model that explains our data, we build a valency dictionary that almost completely dispenses with human intervention. Our procedure is used here for French, but can be used for any other language for which sufficient written data exists.

Keywords: word sense induction, valency, computational lexicography

Table des matières

Liste des tableaux

Liste des figures

Entrées de notre dictionnaire

Liste des sigles et abréviations

DEM	*Dictionnaire électronique des mots*
FTB	*French Treebank*
HOPS	*Honest Parser of Sentences*
LAS	score d'attachement étiqueté, de *labeled attachment score* en anglais
L*efff*	*Lexique des formes fléchies du français*
LSTM	réseaux de longue mémoire à court terme, de l'anglais *long short-term memory networks*
LVF	*Les verbes français*
MCL-WiC	désambiguisation multilingue et interlingue de mots en contexte, de l'anglais *Multilingual and Cross-lingual Word-in-Context Disambiguation*
TLFi	*Trésor de la langue française informatisé*
RNN	réseaux de neurones récurrents, de l'anglais *recurrent neural networks*
SVD	décomposition en valeurs singulières, de *singular value decomposition* en anglais
TAL	traitement automatique des langues
UAS	score d'attachement non étiqueté, de *unlabeled attachment score* en anglais
UD	Universal Dependencies
UFeats	caractéristiques universelles, de *universal features* en anglais

Glossaire

Français	**Anglais**
Amortissement	Damping
Analyse en composantes principales	Principal component analysis
Désambiguïsation lexicale	Word sense disambiguation
Désambiguïsation multilingue et interlingue des mots en contexte	Multilingual and Cross-lingual Word-in-Context Disambiguation
F_1 B^3	BCubed F_1
Induction de sens des mots	Word sense induction
Méthode basée sur les connaissances	Knowledge-based method
Plongement lexical	Word embedding
Partie du discours	Part of speech
Porte de mémoire	Forget gate
Préférence	Preference
Propagation d'affinité	Affinity Propagation
Recherche en grille	Grid search
Référentiel	Baseline
Regroupement	Clustering
Regroupement agglomératif	Agglomerative Clustering
Réseaux de neurones récurrents	Recurrent neural networks
Réseaux de longue mémoire à court terme	Long short-term memory
Seuil de distance	Distance threshold
Transformeurs	Transformers

Remerciements

Ma maîtrise a été un réel plaisir. Pour cela, j'ai plusieurs personnes à remercier.

Tout d'abord, mon directeur de recherche, François Lareau, qui a su me donner exactement le soutien dont j'avais besoin pour avancer dans la bonne humeur et l'enthousiasme.

Ensuite, mes collègues de l'OLST, drôles, brillants et chaleureux, qui ont fait de l'université un environnement de travail enviable.

Laura Kallmeyer, qui m'a accueillie dans son laboratoire de recherche à Düsseldorf pour un séjour de six mois que je n'oublierai jamais.

Et enfin, les personnes et organismes qui m'ont soutenue financièrement : Mme Xiaobo-Ren, pour son généreux appui financier, le Conseil de recherches en sciences humaines, qui m'a permis de me consacrer entièrement à mes études, ainsi que Mitacs Globalink et les bourses d'études supérieures du Canada - Suppléments pour études à l'étranger Michael-Smith, qui ont financé mon séjour en Allemagne.

Chapitre 1

Introduction

Les dictionnaires de valence tels que *Verbənet* (Danlos *et al.*, 2016), *Dicovalence* (van den Eynde et Mertens, 2003), *Les verbes français* (LVF) (Hadouche et Lapalme, 2010), *Dictionnaire électronique des mots* (DEM) (Dubois et Dubois-Charlier, 2010) ou *Lexique des formes fléchies du français* (L*efff*) (Sagot, 2010) sont utiles dans de nombreuses applications de traitement du langage naturel, par exemple pour la génération de langage naturel basée sur des règles. Ce type de dictionnaire indique précisément comment un prédicat exprime syntaxiquement ses arguments, en incluant des informations sur la partie du discours, la préposition ou le cas sélectionnés. Cependant, la manière dont un mot exprime ses arguments peut changer de manière significative en fonction de son sens. Par exemple, le verbe *changer* nécessite un objet direct lorsqu'il signifie 'modifier', comme dans *La discussion a changé mon opinion sur la question*, mais avec le sens 'devenir différent', comme dans *Elle a complètement changé en vieillissant*, il n'y a pas d'objet du tout, selon *WordNet* (Miller, 1992) (traduction libre). Par conséquent, un dictionnaire de valence doit distinguer au moins les principaux sens d'un lemme. La construction manuelle de ce type de ressource est cependant très coûteuse en temps et en ressources, et nécessite un personnel hautement qualifié.

Ce mémoire a donc les objectifs suivants ;

1. Extraire le sens des verbes à partir de données brutes, sans utiliser de base de sens externe, et avec une méthode qui peut s'appliquer à plusieurs langues ;
2. Extraire automatiquement les cadres syntaxiques des verbes ;
3. Créer un dictionnaire de valence lisible à la fois par les humains et par les machines, qui distingue bien les différents sens à l'aide d'exemples, et disponible librement [1].

1. Le dictionnaire final peut être consulté à l'adresse `https://github.com/NaimaHassert/dictionnaire-valence`

Nous avons divisé ce travail en deux grandes étapes : l'identification automatique de la polysémie, et l'extraction automatique des cadres syntaxiques. Ces deux tâches seront abordées dans les chapitres 2 et 3. Avant d'entrer dans les détails, nous examinerons dans ce chapitre plusieurs lexiques combinatoires qui ont été créés automatiquement ou semi automatiquement, en soulignant à la fois leurs points forts et les aspects qui nécessitent selon nous une amélioration. Ceci nous permettra de déterminer les étapes à suivre pour réaliser notre dictionnaire.

Le chapitre 2 fera une revue de la littérature des techniques d'induction de sens, puis détaillera nos expériences et nos résultats. Le chapitre 3 sera consacré à l'évaluation des différents analyseurs syntaxiques automatiques afin de déterminer le plus performant dans le cadre de notre tâche, ainsi qu'à la description de notre méthode pour extraire les cadres syntaxiques. Le chapitre 4 présentera le dictionnaire que nous avons créé ainsi qu'une discussion sur les aspects à améliorer dans des travaux futurs, pour finalement conclure le mémoire avec le chapitre 5.

1.1 Ressources lexicales existantes

La première chose que nous avons faite avant de commencer la construction de notre dictionnaire a été d'observer les travaux similaires qui ont été réalisés dans le passé. Cela nous a permis de déterminer les forces et les faiblesses de chacun d'entre eux, ainsi que de comparer les différentes méthodes utilisées. C'est ainsi que nous avons choisi les caractéristiques que nous voulions que notre dictionnaire possède, ainsi que la marche à suivre pour le réaliser. Dans ce chapitre, nous commençons par une revue de la littérature, avant de détailler les étapes nécessaires à la création automatique d'un dictionnaire de valence.

Les ressources lexicales indiquant la valence des verbes sont traditionnellement construites manuellement par des linguistes. Toutefois, avec le développement des corpus et des techniques d'analyse syntaxique pour le traitement automatique des langues (TAL), des méthodes automatiques d'induction de ressources lexicales ont également été développées. Voici une liste (non exhaustive) des ressources en français créées automatiquement ou semi automatiquement au cours des 15 dernières années.

SYNLEX

SYNLEX (Falk *et al.*, 2007) est un lexique construit à partir du *Lexique-Grammaire* (Gross, 1975) et adapté pour être utilisable en TAL. Il contient 5 244 verbes, 19 127 paires associant

un verbe à son cadre de sous-catégorisation et 726 cadres de sous-catégorisation différents.

LexSchem

LexSchem (Messiant *et al.*, 2008) est le premier lexique de sous-catégorisation verbale pour le français acquis de façon automatique. Ce travail est très similaire à celui que nous cherchons à réaliser, puisque les auteurs extraient les compléments syntaxiques à partir d'un corpus non annoté, en utilisant un analyseur syntaxique, et en français qui plus est. Ils procèdent comme suit :

1. **Prétraitement**. Les tokens du corpus sont étiquetés et lemmatisés à l'aide de *TreeTagger* (Schmid, 1995). Dans notre cas, nous avons décidé de ne pas prétraiter notre corpus, considérant la bonne performance des analyseurs syntaxiques modernes et le temps supplémentaire que cette étape aurait demandé.
2. **Extraction de cadres syntaxiques**. Le corpus prétraité est analysé syntaxiquement avec *Syntex* (Bourigault *et al.*, 2005) pour extraire les compléments des verbes. Les auteurs soulignent que *Syntex* ne distingue pas les arguments des ajouts, problème qui a été partiellement résolu avec les analyseurs modernes comme *Stanza* (mais voir discussion 5).
3. **Filtrage des résultats**. Étant donné que le processus est entièrement automatique, une étape de filtrage est nécessaire pour supprimer le bruit inévitablement engendré dans les étapes précédentes. Pour ce faire, les auteurs utilisent la méthode d'estimation de vraisemblance maximale. Le nombre d'occurrences de chaque cadre syntaxique est calculé, puis divisé par le nombre d'occurrences total du verbe en question : il s'agit de la fréquence relative. Si la fréquence relative d'un cadre syntaxique est plus basse qu'un nombre déterminé empiriquement, alors le cadre est considéré comme erroné.

Le lexique final comprend 3 268 types de verbes (un verbe et sa forme réflexive sont considérés comme deux verbes différents). Une évaluation par rapport à 20 verbes du *Trésor de la langue française informatisé* (TLFi) a révélé un score F_1 de 65 %. À noter que le *LexSchem* ne prend pas en compte les changements éventuels des cadres syntaxiques selon les différents sens des verbes, et qu'au meilleur de nos connaissances, il n'est pas disponible librement.

TreeLex

TreeLex (Kupść et Abeillé, 2008) est un lexique de sous-catégorisation extrait automatiquement à partir du *French Treebank* (FTB), un corpus d'environ 20 000 phrases annotées syntaxiquement et lexicalement, pour un total d'environ 2 000 lemmes verbaux. Le processus d'annotation du FTB est semi-automatique, c'est-à-dire qu'il a été annoté dans un premier temps par des outils automatiques, puis corrigé manuellement.

Les auteurs de *Treelex* ont d'abord extrait les cadres syntaxiques des verbes qui se situaient dans la clause principale, puisque ceux-ci ont tendance à être accompagnés de tous leurs arguments. Les sujets manquants ont été rajoutés aux verbes aux formes infinitive ou impérative, et les compléments manquants ont été rajoutés aux subordonnées. Les verbes au passif ont également été convertis à l'actif.

Notre travail se différencie principalement du leur en raison de la quantité de données disponibles (nous avons également 2 000 lemmes verbaux, mais 20 000 exemples de chacun de ces lemmes, plutôt que 20 000 exemples au total). De plus, nous ne voulons aucunement modifier la structure des phrases. Nous gardons toutefois l'idée de ne conserver que les verbes dans les clauses principales, ce que nous pouvons faire sans trop affecter la qualité des données si nous considérons la quantité que nous avons.

Dicovalence

Le *Dicovalence* (van den Eynde et Mertens, 2003) est un dictionnaire de la valence de plus de 3700 verbes du français construit manuellement. Il est basé sur l'approche pronominale (van den Eynde et Blanche-Benveniste, 1978). La ressource est disponible librement dans un format utilisable en TAL [2]. Chaque entrée correspond à un usage d'un verbe, et comprend les liens syntaxiques et la partie du discours de chacun des arguments, la réalisation pronominale de chaque dépendant, un exemple du verbe en contexte, des traductions du verbe en anglais et en néerlandais, l'auxiliaire qu'il sélectionne ainsi que ses réalisations au passif. En tout, cette ressource comprend 8 000 entrées.

Lexique des formes fléchies du français (L*efff*)

Le L*efff* (Sagot, 2010) a commencé à être développé en 2003, dans l'optique de créer le premier lexique syntaxique pour le français libre d'utilisation et à large couverture. Il se concentrait toutefois au départ sur la morphologie des verbes. C'est par la suite que le L*efff*

2. `https://www.ortolang.fr/market/lexicons/dicovalence/v1`

s'est étendu à toutes les parties du discours et est devenu un lexique syntaxique en plus d'être un lexique morphologique. Les informations syntaxiques originales ont été intégrées à la main, puis étendues à l'aide de techniques automatiques et de corrections et d'ajouts manuels. La version 3.4 du L*efff* [3] contient 5 733 occurrences de verbes, dont 5 444 sont uniques. La grande majorité de ces verbes ne comportent donc qu'un sens, pour une moyenne de 1,05 sens par verbe. En comparaison, à la suite de notre étape de regroupement de sens telle que décrite dans la section 2, nous obtenons en moyenne 2,23 sens par verbe. En outre, seuls quelques verbes ont une indication du sens associé : le verbe *aimer*, dont deux des entrées sont suivies respectivement de la traduction en anglais 'like' et 'love', le verbe *demander* qui indique une distinction entre 'ask' et 'wonder', le verbe *assister* avec une distinction entre 'assist' et 'attend', et le verbe *voler* avec une distinction entre 'fly' et 'steal'. C'est une lacune à laquelle nous tenterons de remédier partiellement dans ce mémoire ; nous tenterons de mieux indiquer le sens associé aux cadres syntaxiques, non pas à l'aide de traductions comme le L*efff*, mais à l'aide d'exemples. Sur le plan du recoupement, toutefois, la grande majorité des verbes que nous avons recueillis sont également présents dans le L*efff*, à l'exception de verbes relativement peu communs dans le langage courant comme *cambrer*, *banquer*, *moyenner*, *ouvrer*, etc.

LexFr

LexFr (Rambelli *et al.*, 2016) est un lexique décrivant les propriétés sémantiques et syntaxiques de 2 493 verbes du français (ainsi que 7 939 noms et 2 628 adjectifs) extraites automatiquement à partir d'un corpus de 90 millions de mots. Ce lexique a été créé à partir d'un modèle existant, soit le *LexIt* (Lenci *et al.*, 2012), pour les verbes, noms et adjectifs italiens. Ce modèle utilise un analyseur syntaxique automatique, adapté par les auteurs pour extraire les relations syntaxiques, gérer les phénomènes problématiques comme les phrases au passif et l'identification des antécédents des pronoms relatifs, et extraire d'autres informations pertinentes comme la présence du pronom réflexif *se*. Le filtrage du résultat de cette analyse syntaxique se fait à partir d'une liste de cadres syntaxiques, à laquelle le cadre de chaque phrase est comparé. Le *LexFr* intègre non seulement ces informations syntaxiques, mais également des informations sémantiques. Les cadres syntaxiques trouvés à l'aide des étapes précédentes sont donc additionnés d'informations sémantiques tirées du WordNet français, telles que les classes sémantiques et les lexèmes les plus typiques des arguments. Le résultat pour 20 verbes choisis automatiquement a été évalué contre le *LexSchem*, pour un score F_1 de 70 %.

3. `https://www.labri.fr/perso/clement/lefff/`

UDLex

UDLex (Rambelli *et al.*, 2017) est la première tentative de construire un lexique de sous-catégorisation qui repose sur le cadre d'annotation de UD, d'une part afin de vérifier si ce cadre est suffisant pour décrire le cadre syntaxique des verbes de certaines langues, et d'autre part afin de comparer la structure syntaxique des verbes de plusieurs langues. Dans cet article, les auteurs expérimentent sur l'anglais, l'italien et le français. Plutôt que d'analyser syntaxiquement un corpus à l'aide d'un analyseur automatique, les auteurs ont choisi d'utiliser les corpus arborés annotés en UD et fournis dans le cadre de la tâche CoNLL 2017 [4]. Le corpus français comprenait approximativement 483 000 tokens. Pour identifier les cadres syntaxiques les plus significatifs, les auteurs font une estimation de la vraisemblance maximale, une méthode statistique très simple qui permet de trouver les paramètres les plus vraisemblables d'un modèle de probabilité pour expliquer des données observées. Les résultats en français sont ensuite comparés au Dicovalence, ce qui donne un score F_1 de 47 %, soit le score le plus faible parmi les trois langues évaluées.

Verbənet

Verbənet (Danlos *et al.*, 2016) est une adaptation du *VerbNet* (Kipper *et al.*, 2006), une ressource pour l'anglais, vers le français. Cette ressource se base sur les classes de Levin, un système qui classifie les verbes sémantiquement similaires en fonction de leurs alternances syntaxiques et des critères sémantiques et morphologiques. Il indique également des rôles thématiques et une décomposition sémantique.

Pour adapter cette ressource pour le français, les auteurs font correspondre les classes de *VerbNet* avec les informations syntaxiques et sémantiques encodées dans le LVF et dans le *Lexique-Grammaire* (Gross, 1975). En 2016, 5 000 emplois de verbes pour environ 3 000 lemmes verbaux étaient recensés dans cette ressource (Danlos *et al.*, 2016).

1.2 Induction automatique d'un dictionnaire de valence : les étapes

Après avoir étudié les travaux précédents, nous avons pu déterminer les étapes suivantes pour réaliser un dictionnaire de valence.

4. `http://universaldependencies.org/conll17/`

1. **Choix du corpus**. Le choix du corpus est la première étape de notre travail. Ce corpus doit être :

(a) grand, pour avoir un nombre suffisant d'occurrences pour chaque verbe ;

(b) balancé, pour que les fréquences de chaque verbe soient à peu près équivalentes ; et

(c) représentatif du langage courant, pour éviter les usages dépassés des verbes et pour détecter les usages nouveaux.

   Dans notre cas, nous avons décidé de tirer nos phrases à analyser de Sketch Engine [5]. Ce moteur de recherche tire des phrases du web et les analyse sommairement afin d'identifier les parties du discours des mots les composant. De cette façon, nous obtenons un corpus varié et moderne, et nous nous assurons d'avoir un nombre suffisant d'exemples pour chacun des verbes.

   Pour extraire ces exemples, nous avons tout d'abord cherché tous les lemmes différents du moteur de recherche et les avons ordonnés selon leur fréquence. Nous avons ensuite filtré ces lemmes selon leur partie du discours pour ne conserver que les verbes. Des 7 642 verbes résultants, nous avons gardé les 2 000 plus fréquents. Au-delà de ce nombre, nous tombons souvent sur de faux lemmes comme *bla* (2 062^{e} "verbe" le plus fréquent) ou *however* (7 642^{e} "verbe" le plus fréquent). Des 2 000 mots obtenus, nous avons supprimé ceux qui avaient été faussement étiquetés comme étant des verbes (par exemple, *d*, *t*, *qu*, *etc.*, *ca*, *quant*, *tandis*, *est-à-dire*, *fr*, *cest*). À ce nombre, nous avons rajouté 18 verbes qui étaient présents dans un de nos jeux de données d'évaluation (celui de la tâche de MCL-WiC), mais qui n'étaient pas dans notre liste originale, afin d'utiliser au maximum ce petit jeu de données. Au total, nous arrivons à 1 958 verbes. Par la suite, nous avons extrait, pour chacun de ces 1 958 verbes, 20 000 phrases les contenant (les verbes moins fréquents ont parfois moins d'exemples disponibles). Ce sont ces phrases que nous avons ensuite classées par sens et analysées syntaxiquement.

2. **Induction des sens des verbes (chapitre 2).** Cette étape est l'étape novatrice de notre travail, et par conséquent, nous n'avons pas pris notre inspiration des travaux cités précédemment. Comme l'un des objectifs de notre travail est de tester l'utilité des plongements lexicaux contextuels pour l'élaboration d'un dictionnaire de valence, nous avons testé plusieurs modèles de langue de type Tranformeurs pour extraire ces plongements lexicaux et avons opté pour une technique de regroupement afin de distinguer les différents sens des verbes.

5. `https://www.sketchengine.eu/`

(a) **Choix d'un algorithme de regroupement des données (section 2.3.2).** Nous avons testé trois des algorithmes de regroupement de données les plus connus et qui peuvent être facilement utilisables à travers la librairie Python *scikit-learn*.

(b) **Extraction des plongements lexicaux (section 2.3.1).** Nous avons testé trois modèles de langue récents, c'est-à-dire qui ont été fournis au public à partir de 2020. Nous avons pris soin de tester à la fois des modèles qui ont été entraînés sur des données de plusieurs langues à la fois (modèles dits multilingues) et un modèle ayant été entraîné uniquement sur le français (modèle monolingue) afin de comparer leurs performances.

(c) **Choix de la méthode d'évaluation (section 2.4).** Les cadres d'évaluation pour l'induction de sens sont rares, particulièrement pour le français. Nous nous sommes donc évalués sur deux cadres d'évaluation différents : un cadre qui vise à évaluer la capacité des systèmes à dire si deux mots dans deux contextes différents ont le même sens ou non, et un cadre destiné à évaluer la capacité des systèmes à étiqueter sémantiquement des mots à partir d'une banque de sens préétablie. Pour ce dernier, nous avons dû trouver une mesure qui permet de faire fi des catégories préétablies, puisque par définition, la tâche d'induction de sens ne repose sur aucune banque de sens.

(d) **Découverte des meilleurs paramètres pour l'algorithme de regroupement (section 2.5).** Pour trouver les meilleurs paramètres et la meilleure combinaison de modèle de langue et d'algorithme de regroupement, nous avons testé toutes les combinaisons possibles de modèles et d'algorithmes, et effectué une recherche en grille pour tester les différents paramètres. Chaque fois, nous avons comparé nos résultats avec les cadres d'évaluation, ce qui nous a permis de déterminer quelle combinaison nous donnait les meilleurs scores.

3. **Extraction des cadres syntaxiques**. Nous avons décidé d'utiliser UD pour notre analyse syntaxique. Ce cadre d'annotation est largement utilisé dans la communauté de TAL et c'est le cadre qu'utilisent les analyseurs automatiques parmi les plus performants de nos jours. Nous avons ensuite suivi les étapes suivantes :

(a) **Choix d'un analyseur syntaxique (section 3.4)**. Nous avons évalué quatre analyseurs syntaxiques sur les données d'évaluation de UD du français, en prenant compte non seulement du score final (sachant que les données de référence sont imparfaites) mais également de la rapidité d'exécution, puisque nous devons analyser des millions de phrases.

(b) **Choix des éléments à conserver.** Pour construire les cadres syntaxiques, nous avons décidé de ne considérer que les verbes qui sont la racine des phrases (indiquées `root`

en UD), puisque ces verbes risquent davantage d'être accompagnés de tous leurs arguments. Parmi les dépendants de ces verbes, nous n'avons gardé que les éléments qui correspondent à de réels arguments, et non à des ajouts.

(c) **Filtrage (section 3.6).** Les résultats ont certainement du bruit, dû à des erreurs d'étiquetage de parties du discours et d'analyse syntaxique. En raison de son efficacité et de sa simplicité, nous avons décidé d'utiliser une méthode d'estimation de vraisemblance maximale, ce qui revient à déterminer un simple seuil sur les fréquences relatives des candidats de cadres syntaxiques.

(d) **Évaluation (section 3.6.2).** Comme les cadres théoriques varient beaucoup et qu'une ressource lexicale créée automatiquement et tenant compte du sens des verbes comme celle que nous souhaitons créer n'existe pas encore, nous ne voulons pas comparer le dictionnaire complet avec un autre. Toutefois, nous pouvons déterminer le seuil qui permet d'extraire des cadres les plus pertinents possibles à l'aide d'une ressource existante. Le seuil a donc été déterminé en comparant les cadres syntaxiques obtenus à ceux contenus dans une ressource créée manuellement.

4. **Construction du dictionnaire.**

   Nous souhaitons créer une ressource libre d'accès et aisément lisible à la fois pour les ordinateurs et pour les humains. Il nous faut donc déterminer un format, que nous présenterons dans le chapitre 4.

Chapitre 2

Identification automatique de la polysémie

2.1 Désambiguïsation lexicale

La désambiguïsation lexicale est une tâche du TAL qui vise à déterminer automatiquement le sens d'un mot donné. En effet, les langues sont pleines de mots ambigus : en français, par exemple, le mot-forme *souris* peut faire référence au rongeur qui aime le fromage, à la souris d'ordinateur, ou même au verbe *sourire* accordé au présent, à la deuxième personne du singulier.

Les humains arrivent naturellement à désambiguïser les mots selon leur contexte : ainsi, une personne confrontée à la phrase *Ma souris est brisée, je dois en acheter une autre* n'aura aucun mal à conclure qu'on parle ici d'un objet, et non d'un animal. Mais quand on s'attarde un peu plus longuement sur la question, on se rend compte qu'il ne s'agit pas d'une tâche triviale, loin de là. Non seulement il nous faut des connaissances syntaxiques pour reconnaître que *souris* est un nom étant donné sa position dans la phrase, mais également des connaissances sur les mots-formes environnants, qui nous permettent par exemple de savoir que le verbe *briser* s'applique à un objet, et non à un animal. Certaines connaissances du monde peuvent également nous être utiles : si notre animal de compagnie meurt, pourquoi dirait-on qu'il faut en acheter un autre ?

La plupart des modèles de langue, à l'exception possible des modèles dernier cri comme *ChatGPT*, n'ont pas encore atteint ce niveau de connaissances et de sophistication. Il nous faut encore leur donner des instructions très précises pour qu'elles puissent accomplir correctement la tâche de désambiguïsation lexicale.

La désambiguïsation lexicale est une étape essentielle dans de nombreuses applications du TAL, telles que :

1. la traduction automatique, où un mot dans une langue peut avoir plusieurs traductions différentes dans une autre ;
2. la recherche d'information, où les requêtes de recherche contiennent souvent des mots ambigus ;
3. l'extraction d'information, où nous voulons récupérer automatiquement des informations spécifiques liées à un sujet précis ;
4. et la lexicographie, où nous voulons souvent obtenir des informations lexicales spécifiques à un sens d'un lemme donné.

Une façon courante d'aborder cette tâche consiste à utiliser une méthode basée sur les connaissances ou une méthode supervisée. Les méthodes fondées sur la connaissance s'appuient fortement sur des ressources existantes telles que *WordNet* (Miller, 1992), *BabelNet* (Navigli et Ponzetto, 2012) ou d'autres dictionnaires, et utilisent le contenu de ces ressources pour le comparer aux données disponibles et en déduire le sens du mot. Les méthodes supervisées s'appuient plutôt sur des données annotées en fonction du sens, qui sont ensuite utilisées pour annoter les données brutes. La plupart des méthodes de pointe sont hybrides, c'est-à-dire qu'elles combinent les caractéristiques des méthodes fondées sur les connaissances et des méthodes supervisées (Bevilacqua *et al.*, 2021).

Dans le contexte de la création d'un dictionnaire, cependant, les méthodes basées sur les connaissances ou supervisées ne sont pas nécessairement les plus appropriées pour identifier le sens d'un mot ambigu, pour les raisons suivantes :

1. **Les sens répertoriés dans les principales ressources lexicales sont souvent trop fins.** Une ressource lexicale populaire dans le domaine du TAL est *WordNet*, un dictionnaire électronique de l'anglais basé sur des *synsets*, c'est-à-dire des ensembles de lexèmes de la même partie du discours et qui partagent approximativement le même sens. Si l'on recherche un mot dans *WordNet*, on obtient tous les *synsets* qui le contiennent. Dans le cas de *change* (*changer* en français), par exemple, il y a 10 *synsets* liés au nom *change*, et 10 synsets liés au verbe *change*. Son homologue multilingue, BabelNet (Navigli et Ponzetto, 2012), est le résultat de la fusion de *WordNet* et de *Wikipédia*, où les *synsets* sont fournis en partie par les traductions manuelles fournies par *Wikipédia* et en partie par un système de traduction automatique. Cependant, il a été souligné que les sens de *WordNet* sont très détaillés, au point que l'accord interannotateurs lors de l'utilisation

de l'inventaire de *WordNet* est d'environ 70 %, seulement 5 % de plus que le référentiel standard, qui consiste à annoter chaque mot avec son sens le plus fréquent (Raganato *et al.*, 2017).

2. **La plupart des ressources sont basées sur l'anglais.** Les systèmes de désambiguïsation lexicale s'appuient fortement sur des données annotées par sens. Ce type de données existe en anglais, principalement grâce à *SemCor* (Miller *et al.*, 1993), qui est annoté sur la base de *WordNet*. Cependant, l'annotation sémantique manuelle étant très coûteuse en temps et en ressources, ces données sont très rares, voire inexistantes, pour les langues autres que l'anglais. Par conséquent, les ressources lexicales dans d'autres langues sont souvent dérivées des ressources anglaises, comme *Europarl* (Koehn, 2005), un corpus annoté selon les informations sémantiques de *BabelNet* (qui est lui-même dérivé en partie de *WordNet*). S'appuyer sur ces ressources peut donc être trompeur si l'on cherche à effectuer une désambiguïsation lexique pour une langue comme le français, par exemple.
3. **Le recours à des ressources externes empêche la découverte de nouveaux sens.** Les ressources lexicales ont l'inconvénient d'être coûteuses à créer et à maintenir, comme expliqué précédemment. Cependant, avec l'évolution constante de la langue, ces ressources manuelles risquent de devenir rapidement obsolètes.

2.2 Induction de sens des mots

Lorsque la signification d'un mot doit être déterminée sans utiliser de ressource externe, cela est appelé *induction* ou *discrimination* de sens des mots. Ces méthodes sont particulièrement utiles dans les situations où l'acquisition de connaissances est ralentie ou limitée en raison d'un manque de ressources, de temps ou de compétences pour collecter, traiter et intégrer de nouvelles informations dans un système existant. Contrairement aux méthodes qui utilisent des ressources externes, l'induction de sens ne repose que sur des données brutes et non annotées. L'objectif n'est pas d'attribuer une signification à un mot en particulier, mais plutôt de détecter le nombre de significations en partant du principe que deux occurrences d'un mot ont le même sens si elles se produisent dans des contextes similaires.

Les travaux les plus notables dans ce domaine sont les suivants :

Regroupement de contextes

Cet algorithme développé par Schütze (1998) considère que les différents sens d'un mot sont représentés par des regroupements de contextes similaires. Concrètement, chaque mot est

représenté sous forme d'un vecteur dont les composantes sont le nombre d'occurrences des autres mots dans son contexte (ce contexte peut être une phrase, un paragraphe, ou toute autre portion de texte). L'algorithme original utilise des vecteurs construits à partir de co-occurrences de second ordre, c'est-à-dire que les vecteurs des mots dans le contexte du mot ambigu sont eux-mêmes construits à partir de leur propre contexte. Ces vecteurs de contexte peuvent ensuite être regroupés en fonction de leur similarité. Chaque groupe est représenté par la moyenne de tous les vecteurs de ce groupe, appelé le *centroïde*. Cette méthode est la plus proche de celle choisie dans le cadre de ce mémoire.

Regroupement de mots

Cet algorithme a été développé par Lin (1998) et se base sur l'analyse syntaxique des contextes pour identifier les mots similaires à un mot cible. Les contextes sont représentés sous forme de triplets composés du mot cible, d'un dépendant syntaxique et de la relation syntaxique entre eux. Les triplets communs à plusieurs mots sont utilisés pour calculer leur similarité. Pour regrouper les mots en différents sens, l'algorithme utilise un arbre de regroupement dont les nœuds fils du nœud principal représentent les différents sens du mot. Cette méthode diffère des approches basées sur les cooccurrences de second ordre en se focalisant sur les dépendances syntaxiques des mots.

Graphes de cooccurrences

En 2004, Véronis a présenté *HyperLex* (Véronis, 2004). Selon l'auteur, le regroupement de vecteurs peut présenter un inconvénient majeur, à savoir qu'il peut exclure des sens de mots moins courants qui peuvent être considérés comme du bruit par l'algorithme, même si ces sens ne sont pas rares pour un locuteur moyen. Pour pallier ce problème, l'auteur propose une alternative qui consiste à construire un graphe dont les nœuds sont des mots, et où les relations entre ces nœuds sont établies en fonction de leurs cooccurrences dans un contexte donné. Cette approche permet de former des "petits mondes", c'est-à-dire des groupes de mots fortement liés entre eux qui correspondent potentiellement à un sens du mot ambigu et qui sont tous liés d'une manière ou d'une autre.

Réseaux de neurones récurrents

Les réseaux de neurones récurrents, de l'anglais *recurrent neural networks* (RNN), sont souvent utilisés dans le TAL pour leur capacité à prendre en compte les relations séquentielles entre les données. Cependant, ces réseaux souffrent du problème de la perte de gradient, qui limite leur efficacité pour les séquences longues. Ce problème est causé par la diminution de

la valeur du gradient de l'erreur (mesure de la direction dans laquelle les poids doivent être ajustés pour minimiser l'erreur) lorsqu'on se propage à travers un grand nombre de couches dans un réseau profond. Dans le cas des RNN, qui utilisent l'état caché à chaque étape pour la rétropropagation, ce problème est particulièrement important et limite leur utilisation à des séquences courtes.

Les réseaux de longue mémoire à court terme, de l'anglais *long short-term memory networks* (LSTM) (Hochreiter et Schmidhuber, 1997) ont été proposés pour résoudre ce problème. Cette variante des RNN utilise des portes de mémoire pour contrôler l'écoulement de l'information, ce qui permet de conserver les informations importantes et d'éviter la perte de gradient. Cependant, les LSTM nécessitent plus de ressources et de temps pour être formés que les RNN standards et ne sont pas aussi efficaces sur le plan matériel en raison de leur nature linéaire. De plus, ils ne disposent toujours pas d'une mémoire importante malgré leur amélioration par rapport aux RNN standards.

Transformeurs

Vaswani *et al.* (2017) ont présenté les transformeurs, qui ont révolutionné le domaine en utilisant un mécanisme d'attention leur permettant de traiter toute l'entrée simultanément. Cela a considérablement réduit la durée d'entraînement et permis d'obtenir des résultats de pointe en TAL. Les modèles transformeurs préentraînés sur des ensembles de données massifs peuvent être facilement téléchargés et ajustés à l'aide de la plateforme Hugging Face [1].

2.3 Méthode

2.3.1 Plongements lexicaux

Nous avons utilisé trois modèles de langue différents pour nos expériences, tous téléchargés depuis Hugging Face. Nous avons utilisé un modèle monolingue pour le français, *CamemBERT*, et deux modèles multilingues, *XLM-RoBERTa* et *T5*. Des études ont démontré que les modèles monolingues donnent de meilleurs résultats que les modèles linguistiques multilingues tels que *mBERT* (Martin *et al.*, 2020). Cependant, en 2021, *XLM-RoBERTa* a montré des résultats impressionnants sur la tâche 2 de SemEval 2021, portant sur la désambiguïsation de mots en contexte (Martelli *et al.*, 2021), y compris pour le français ; nous avons donc également testé ce modèle. Nous avons également essayé un autre modèle de

1. `https://huggingface.co/`

langue multilingue, *T5*, créé par Google. Nous avons utilisé la version “large” de chaque modèle et obtenu nos plongements lexicaux contextualisés en calculant la moyenne de toutes les couches cachées.

CamemBERT

(Martin *et al.*, 2020) est un modèle monolingue construit spécialement pour le français. Son architecture est basée sur celle de *RoBERTa*, une méthode qui s'appuie sur la stratégie de masquage de *BERT* tout en modifiant les hyperparamètres clés et en s'entraînant avec des mini-lots et des taux d'apprentissage beaucoup plus élevés (Liu *et al.*, 2019). *RoBERTa* aurait de meilleures performances de tâche en aval que *BERT*.

CamemBERT est entraîné sur 138 Go de données brutes. Lors de sa sortie en 2020, il a amélioré l'état de l'art pour l'étiquetage des parties du discours, l'analyse syntaxique en dépendance, la reconnaissance des entités nommées et les tâches d'inférence pour le français.

XLM-RoBERTa

XLM-RoBERTa (Conneau *et al.*, 2020) est un modèle de langue multilingue préentraîné sur 2,5 To de données en 100 langues. À sa sortie, il a montré une amélioration très significative par rapport aux modèles multilingues *mBERT* et *XLM-100*, et a obtenu des résultats compétitifs par rapport aux modèles monolingues de pointe, dont *RoBERTa*, en anglais, ce qui a permis aux auteurs de démontrer que les modèles multilingues pouvaient constituer une amélioration par rapport à leur homologue monolingue *BERT*. À notre connaissance, cependant, *CamemBERT* et *XLM-RoBERTa* n'ont pas encore été comparés l'un à l'autre sur une tâche d'induction de sens des mots. Nous vérifierons donc dans ce mémoire si *XLM-RoBERTa* peut performer mieux que son homologue monolingue pour le français.

T5

T5 (Raffel *et al.*, 2020) est une alternative à *BERT*. Au lieu d'avoir une étiquette de classe ou une partie de l'entrée en sortie, comme avec les modèles de style *BERT*, *T5* a uniquement une chaîne de texte en entrée ainsi qu'en sortie. *T5-large*, le point de contrôle que nous avons utilisé, possède 770 millions de paramètres. *T5* a été entraîné sur un ensemble de données contenant 4 langues : l'anglais, le français, l'allemand et le roumain.

2.3.2 Regroupement sémantique

Les méthodes de regroupement visent à découvrir une structure ou des motifs cachés dans un ensemble de données non étiquetées. Certains algorithmes de regroupement traditionnels exigent un nombre prédéterminé de groupes, ce qui peut limiter leur efficacité lorsqu'on cherche à découvrir de nouveaux sens qui ne sont pas répertoriés dans les ressources lexicales existantes. Pour surmonter cette limitation, nous avons testé trois algorithmes de regroupement capables de déterminer automatiquement le nombre optimal de groupes.

Propagation d'affinité

L'algorithme de propagation d'affinité (Dueck, 2009) est un algorithme de regroupement qui permet de trouver des "exemplaires", ou membres représentatifs de chaque groupe, en échangeant des messages entre les données. Il est possible de régler deux paramètres lors de l'utilisation de cet algorithme : l'amortissement et la préférence. L'amortissement affecte la convergence de l'algorithme et permet de régler la vitesse à laquelle les messages sont échangés entre les données. La préférence permet de contrôler le nombre de groupes obtenus en ajoutant du bruit à la matrice de similarité, ce qui influence la création de groupes plus ou moins nombreux.

Regroupement agglomératif

Le regroupement agglomératif (*agglomerative clustering*) (Szekely et Rizzo, 2005) est un type de regroupement qui utilise une approche ascendante. L'algorithme considère d'abord chacune des données comme son propre groupe et agglomère successivement des groupes similaires jusqu'à ce que tous les groupes soient fusionnés en un seul qui contient toutes les données. Dans le module Python *scikit-learn* (Pedregosa *et al.*, 2011), au lieu de spécifier le nombre de groupes, nous pouvons simplement spécifier le *seuil de distance*, c'est-à-dire la mesure de la distance au-delà de laquelle deux groupes ne seront pas fusionnés. Essentiellement, il détermine où couper le dendrogramme (voir la figure 2.1). Nous avons conservé le paramètre de liaison par défaut, à savoir *ward*, et testé différents seuils de distance, allant de 10 à 300 000.

HDBSCAN

HDBSCAN (Campello *et al.*, 2013) est une méthode de regroupement qui étend *DBSCAN* en un algorithme de regroupement hiérarchique. Il est conçu pour gérer un grand nombre de données bruitées, parmi lesquelles peuvent être trouvées des régions de densité plus élevée.

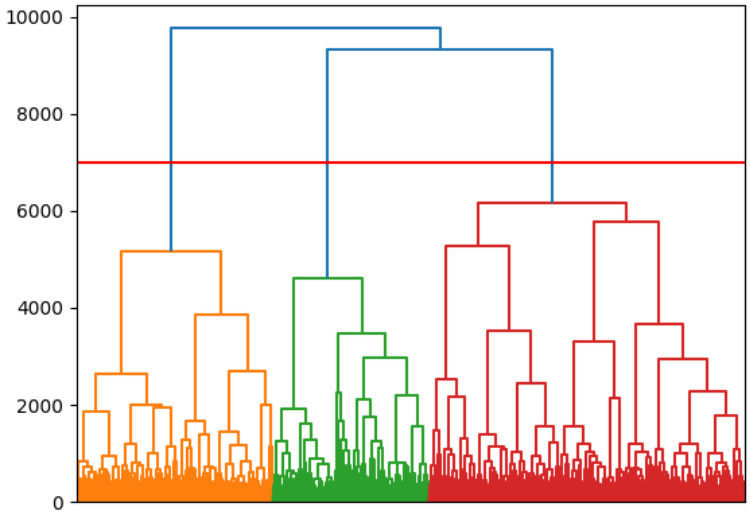

FIGURE 2.1 – Dendrogramme représentant le regroupement de 20 000 plongements contextuels du verbe *adopter* avec l'algorithme de regroupement agglomératif. La ligne rouge horizontale représente le nombre de groupes final (3) avec un seuil de distance de 7 000.

Les algorithmes basés sur la densité ont l'avantage d'être efficaces même lorsque les données n'ont pas été traitées et que les groupes ont une forme inhabituelle. L'algorithme commence par identifier les zones les plus denses de l'espace de données et décide si elles doivent être fusionnées ou maintenues séparées. Pour chaque point, *HDBSCAN* peut calculer un score de probabilité d'appartenance à son groupe. Il peut aussi calculer le score global de la qualité du regroupement.

Il existe trois paramètres principaux que l'on peut ajuster dans *HDBSCAN* : la taille minimale du groupe (la plus petite taille de groupe à considérer comme tel), le nombre minimal d'échantillons (plus la valeur est élevée, plus l'algorithme sera conservateur et plus les données seront considérées comme du bruit) et la méthode de sélection de regroupement (par défaut *eom* (*Excess of Mass*), qui peut être changé en *leaf*, qui a tendance à produire une sélection de groupes plus fine).

2.4 Méthode d'estimation des paramètres

2.4.1 *FrenchSemEval*

FrenchSemEval (Segonne *et al.*, 2019) est un ensemble de données d'évaluation construit spécifiquement pour la désambiguïsation du sens des verbes français. Il a été construit après que les auteurs aient inspecté *Eurosense* (Delli Bovi *et al.*, 2017), un corpus multilingue extrait d'*Europarl* (Koehn, 2005) et automatiquement annoté sémantiquement à partir de l'inventaire multilingue de *BabelNet*. Cette ressource a présenté de bons résultats en ce qui concerne l'accord interannotateurs, et pour l'anglais, les annotations *Eurosense* de haute précision couvrent 75 % des mots du contenu et ont un score de précision de 81,5 %. Comme on peut s'y attendre, cependant, les résultats pour le français sont inférieurs : la couverture est de 71,8 % et la précision est de 63,5 %. En outre, la situation s'aggrave avec les verbes, ce qui est prévisible puisque la désambiguïsation des verbes est réputée être plus difficile (Raganato *et al.*, 2017). Lorsque les auteurs ont examiné les verbes annotés automatiquement dans *Eurosense*, ils se sont rendu compte que la proportion d'entre eux qu'ils jugeaient correcte n'était que de 44 %. Les auteurs ont également constaté que *BabelNet* offre une grande variété de sens par verbe. Dans une étude sur 150 phrases, ils ont découvert que chaque verbe avait en moyenne 15,5 sens selon *BabelNet*, et qu'il était parfois difficile de distinguer ces sens les uns des autres. En résumé, même si *Eurosense* est une ressource utile, elle est basée sur l'anglais et donc moins fiable pour le français. En outre, elle présente des nuances de sens trop subtiles.

En revanche, les auteurs ont observé que dans le *Wiktionnaire* [2], le niveau de granularité des sens est généralement assez naturel et que les distinctions de sens sont faciles à saisir. Ils ont donc décidé d'utiliser les sens du *Wiktionnaire* comme base pour leur annotation manuelle. *FrenchSemEval* est le résultat de cet effort. Il se compose de 3 121 phrases annotées sémantiquement, avec 66 formes verbales différentes, chacune ayant en moyenne 3,83 sens. Tous ces verbes figurent parmi les 2 000 verbes les plus courants identifiés précédemment avec Sketch Engine. Selon l'article, la précision du référentiel pour le sens le plus fréquent est de 30 %.

En guise de données d'entraînement, les données complètes du *Wiktionnaire*, version 2018, sont fournies. 16 935 verbes sont présents dans cette ressource, dont 1 905 d'entre eux étaient également présents dans les verbes les plus fréquents selon Sketch Engine. En moyenne, nous retrouvons deux exemples par sens dans cette ressource.

2. `https://www.wiktionary.org/`

Étant donné le très grand nombre de verbes différents présent dans les données d'entraînement et le temps qui aurait été requis pour déterminer nos paramètres sur ces données, ainsi que le petit nombre d'exemples par sens qu'on trouve dans le *Wiktionnaire*, nous avons pris la décision d'estimer nos paramètres sur *FrenchSemEval*, c'est-à-dire la partie test des données. Cette partie test, conçue manuellement, est selon nous plus près des données que nous allons analyser, et a l'avantage d'être assez petite pour que nous puissions effectuer notre recherche de paramètres dans une durée raisonnable.

2.4.2 MCL-WiC

La tâche de MCL-WiC (Martelli *et al.*, 2021) est la première tâche SemEval qui teste la capacité des systèmes à distinguer les différents sens des mots, sans avoir besoin d'un inventaire de sens préétabli. Dans la sous-tâche multilingue, le système doit décider si deux mots cibles dans deux contextes différents dans la même langue ont le même sens ou non. Les verbes ont été choisis en fonction de leur nombre de sens (au moins trois selon *BabelNet*), et les paires de phrases ont été extraites du corpus parallèle des Nations Unies (Ziemski *et al.*, 2016) ou de *Wikipédia*. Les phrases sélectionnées étaient bien formatées et contenaient suffisamment de contexte sémantique pour permettre une détermination précise du sens des mots cibles.

L'équipe qui a obtenu le meilleur résultat pour la tâche Fr-Fr (MCL@IITK) a atteint 87,5 % de précision. Elle a extrait des plongements lexicaux contextuels optimisés des mots cibles à partir de *XLM-RoBERTa* et les a transmis à une unité de régression logistique. Il faut noter que même si nous avons également testé *XLM-RoBERTa*, nos résultats ne peuvent pas être directement comparés, puisque nous avons évalué nos résultats sur les verbes uniquement (et non sur toutes les parties du discours comme l'équipe MCL@IITK).

2.4.3 Score utilisé pour l'évaluation

Nous mentionnons souvent dans ce travail que nous utilisons le score F_1 B^3, aussi appelé F_1 BCubed ; il a été proposé par **B**agga et **B**. **B**aldwin (1998), d'où le B^3, et popularisé par Amigó *et al.* (2009). En effet, le F_1 standard est conçu pour comparer les données regroupées à l'aide des mêmes étiquettes de groupes, ce qui est utile si les regroupements en question ont une signification spécifique, mais pas autrement. Prenons par exemple le verbe *changer* mentionné dans l'introduction. Si nous avons un groupe A qui doit regrouper tous les verbes qui ont le sens *modifier* et un groupe B qui doit regrouper tous les verbes qui ont le sens

devenir différent, alors les étiquettes de groupe sont importantes. Si tous les verbes placés dans le groupe B avaient dû être dans le groupe A et vice versa, le score F_1 standard sera très faible.

Dans notre cas, cependant, les étiquettes de groupes n'ont aucune signification : quelle que soit l'étiquette, l'important est de regrouper tous les mots qui ont des sens similaires. C'est là que le F_1 B^3 est utile. Au lieu de calculer la précision et le rappel en fonction du nombre de vrais et faux positifs et négatifs dans tous les exemples, ces scores sont calculés pour chaque élément individuellement. La moyenne des nombres obtenus pour chaque exemple dans le document est ensuite calculée pour produire les scores de rappel et de précision pour l'ensemble des données. Les formules pour calculer le rappel et la précision B^3 sont les suivantes :

$$\text{Précision finale} = \sum_{i=1}^{N} w_i \times Precision_i \tag{2.1}$$

$$\text{Rappel final} = \sum_{i=1}^{N} w_i \times Rappel_i \tag{2.2}$$

La formule du score F_1 B^3 ne diffère pas de la formule standard :

$$F_1 = \frac{2 \times \text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \tag{2.3}$$

En d'autres termes, le score F_1 standard est parfait pour évaluer les performances dans une tâche de désambiguïsation du sens des mots, où les groupes sont déjà déterminés. Cependant, dans une tâche d'induction de sens, nous souhaitons évaluer les performances d'un algorithme qui crée des groupes à partir de zéro par rapport à un ensemble de données d'évaluation qui aura nécessairement ses propres étiquettes de groupes. Dans ce cas, le F_1 B^3 doit être utilisé.

2.5 Estimation des paramètres

2.5.1 *FrenchSemEval*

En regroupant les données d'évaluation

Nous avons regroupé les 3 121 phrases de test de *FrenchSemEval* et calculé notre score avec le score F_1 B^3. Les meilleurs résultats pour chaque combinaison d'algorithme de regroupement et de modèle de langue se trouvent dans le tableau 2.1.

Algorithme de regroupement	***T5***	***CamemBERT***	***XLM-RoBERTa***
Propagation d'affinité	14,86 %	**14,87 %**	14,86 %
Regroupement agglomératif	46,02 %	**65,39 %**	56,06 %
HDBSCAN	30,41 %	33,76 %	**35,30 %**

TABLEAU 2.1 – Comparaison des algorithmes de regroupement, en F_1 B^3

Comme nous pouvons le voir, la combinaison de l'algorithme de regroupement agglomératif et de *CamemBERT* est de loin la meilleure pour notre tâche, donnant des résultats plutôt impressionnants pour une méthode non supervisée. En effet, en guise de comparaison, l'équipe *FlauBERT* (Le *et al.*, 2019), en utilisant une combinaison de *CamemBERT* et d'une méthode supervisée, a atteint un score F_1 de 50,02 %. Il faut toutefois garder en tête que les données d'entraînement sur lesquelles cette équipe s'est entraînée sont plutôt limitées, et que les scores que nous présentons sont ceux obtenus sur la partie d'entraînement du jeu de données, non sur la partie test. Néanmoins, nous pouvons voir que notre approche a du potentiel. Le seuil de distance qui a permis à chacun des modèles de langue d'atteindre le meilleur score varie : pour *CamemBERT*, il était de 650 ; pour T5, il était de 100 000 ; et pour *XLM-RoBERTa*, il était de 725.

Les pires résultats ont été obtenus avec l'algorithme de propagation d'affinité. Même en effectuant une recherche en grille avec différentes valeurs d'amortissement et de préférence, nous n'avons pas réussi à atteindre plus de 14,87 % en score F_1 B^3. L'algorithme a obtenu un bon score de précision, mais un très mauvais score de rappel dans chaque combinaison de paramètres, ce qui indique que l'algorithme n'a pas pu généraliser, n'attribuant environ qu'un sens par occurrence du verbe.

HDBSCAN a eu le comportement inverse que l'algorithme de propagation d'affinité : le rappel est généralement bien meilleur que la précision, ce qui indique qu'il a tendance

à n'attribuer qu'un seul sens à l'intégralité des données. Le meilleur résultat avec cet algorithme a été obtenu par *XLM-RoBERTa*, avec une taille minimale des regroupements de 10, un nombre minimal d'échantillons de 2 et la méthode de sélection des classes dite *leaf*. Nous pouvons également noter qu'une énorme quantité de données est considérée comme du bruit par l'algorithme, et que dans presque toutes les configurations de paramètres, la méthode de sélection de regroupement *leaf* donne de bien meilleurs résultats que la méthode de sélection de regroupement *eom*.

En regroupant chaque verbe individuellement

Nous avons regroupé les 20 000 occurrences de chaque verbe présent dans le jeu de données *FrenchSemEval* et nous avons ajouté les cinquante phrases des données d'évaluation. Nous avons ensuite évalué la performance de notre regroupement en utilisant ces phrases d'évaluation.

Il s'avère que le F_1 pourrait ne pas être le meilleur indicateur de la qualité du regroupement dans notre cas. En effet, en augmentant le seuil de distance, le rappel atteint 100 % ou presque, ce qui améliore artificiellement le score. Cela ne signifie pas que le regroupement est de qualité supérieure. Au contraire, cela indique simplement que le regroupement est de moins en moins restrictif, ce qui limite le nombre de groupes créés pour contenir l'ensemble des données. Par exemple, si nous définissons le seuil de distance à 19 000, l'algorithme atteint un score de 67,68 %, qui est supérieur à celui obtenu sur l'ensemble des données. Cependant, en examinant le nombre moyen de groupes, on constate que ce n'est pas un bon regroupement : en moyenne, les occurrences de chaque verbe ne se trouvent que dans un seul groupe.

L'objectif pourrait alors être d'obtenir un nombre de groupes approximativement égal au nombre moyen de groupes pour l'ensemble de données de *FrenchSemEval*, qui est de 3,83. Nous obtenons un nombre moyen de groupes de 3,89 avec un seuil de distance de 6000, avec un score F_1 B^3 de 59,93 %, ce qui reste satisfaisant. Plus le seuil de distance augmente, plus le nombre moyen de groupes diminue, alors qu'au contraire, le score augmente (voir figure 2.2) ; à 7000, le nombre moyen de classes est de 3,13, avec un score de 62,75 %.

2.5.2 MCL-WiC

Après avoir testé notre algorithme sur le jeu de données *FrenchSemEval*, nous étions incertains quant au meilleur paramètre à utiliser, car nous devions trouver un équilibre entre le score et le nombre moyen de groupes. Pour tenter de résoudre cette incertitude, nous avons

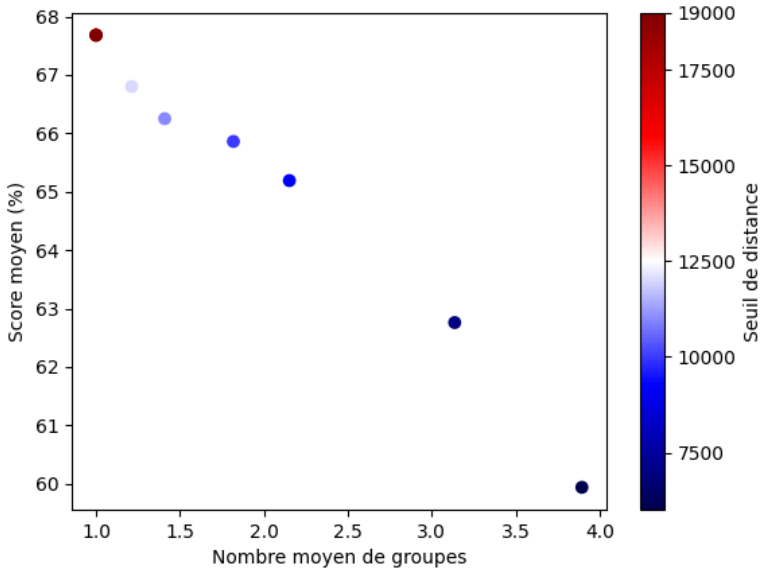

FIGURE 2.2 – Relation entre le score, le nombre de groupes et le seuil de distance lors du regroupement par verbe individuel avec l'algorithme de regroupement agglomératif. Le score est exprimé en F_1 B^3 et calculé à partir du jeu de données *FrenchSemEval* utilisé comme référence.

testé notre algorithme sur un autre jeu de données, MCL-WiC. Nous avons suivi la même méthode que précédemment : nous avons extrait les plongements contextuels de chaque verbe dans les phrases de test à l'aide de *CamemBERT*, puis les avons fusionnés avec les plongements de nos propres données. Ensuite, nous avons regroupé tous les plongements de chaque verbe à l'aide de l'algorithme de regroupement agglomératif, en utilisant trois valeurs de seuil de distance possibles : 6000, 6500 et 7000. Les résultats sont présentés dans le tableau 2.2. Nous avons constaté que le score de précision était le plus élevé avec un seuil de distance de 6000 sur ce jeu de données. Les scores suivent donc une tendance inverse : une augmentation du seuil de distance se traduit par un score diminué pour MCL-WiC, mais amélioré pour *FrenchSemEval*.

2.6 Résultats

Nous avons testé notre combinaison de paramètres trouvés à l'aide du jeu de données de test de *FrenchSemEval* avec un sous-ensemble du *Wiktionnaire*, soit un sous-ensemble du jeu de données d'entraînement. Nous avons choisi 500 verbes au hasard parmi les 1905 qui se

Seuil de distance	MCL-WiC (précision)	*FrenchSemEval* (F_1 B^3)
6000	**61.83 %**	59.93 %
6500	59.92 %	61.3 %
7000	59.54 %	**62.75 %**

TABLEAU 2.2 – Comparaison des résultats sur les jeux de données d'évaluation MCL-WiC et *FrenchSemEval* en fonction de la valeur du seuil de distance utilisé dans l'algorithme de regroupement agglomératif.

retrouvaient à la fois dans le jeu de données d'entraînement et dans nos données extraites nous-mêmes, puis nous avons regroupé nos données de chacun des verbes séparément, en leur injectant celles du jeu de données d'entraînement. Au final, nous obtenons un F_1 B^3 de 58,19 %, avec 54 % de précision, 77 % de rappel et 2,23 sens par verbe en moyenne, contre 4,73 sens dans les données de référence. Ce résultat est un peu plus de 4 % inférieur à celui obtenu sur la partie de test et favorise largement le rappel au détriment de la précision.

2.7 Conclusion

Comparer les résultats des jeux de données MCL-WiC et *FrenchSemEval* ne nous a pas aidé à prendre une décision judicieuse concernant le seuil de distance à utiliser sur nos données. Alors que le score diminue à mesure que le seuil de distance augmente avec MCL-WiC, le score suit la tendance inverse avec *FrenchSemEval*. Étant donné que le jeu de données *FrenchSemEval* est plus grand et plus proche de la tâche que nous voulons accomplir, nous avons décidé de sélectionner le seuil de distance de 7000, qui donnait un résultat satisfaisant pour le jeu de données de test de *FrenchSemEval* (62,75 %) tout en produisant un nombre adéquat de sens par verbe. En évaluant ce paramètre sur un sous-ensemble des données d'entraînement, nous obtenons un score de 58,19 % avec un rappel supérieur à la précision, ce qui peut probablement s'expliquer par le fait que le nombre de sens était plus élevé dans cette partie. Rien ne nous oblige toutefois à conserver ce paramètre : il peut aisément être changé dans notre chaîne de traitement, et ajusté en fonction des besoins. Un seuil de distance plus élevé donnerait probablement un rappel plus élevé, tandis qu'un seuil de distance plus bas favoriserait probablement la précision.

Pour conclure cette section, voici, en guise d'illustration des plongements contextuels obtenus, la représentation en deux dimensions, obtenue à l'aide d'une décomposition en valeurs singulières, de *singular value decomposition* en anglais (SVD), des plongements de mots de

CamemBERT classés par l'algorithme de regroupement agglomératif avec un seuil de distance de 7 000.

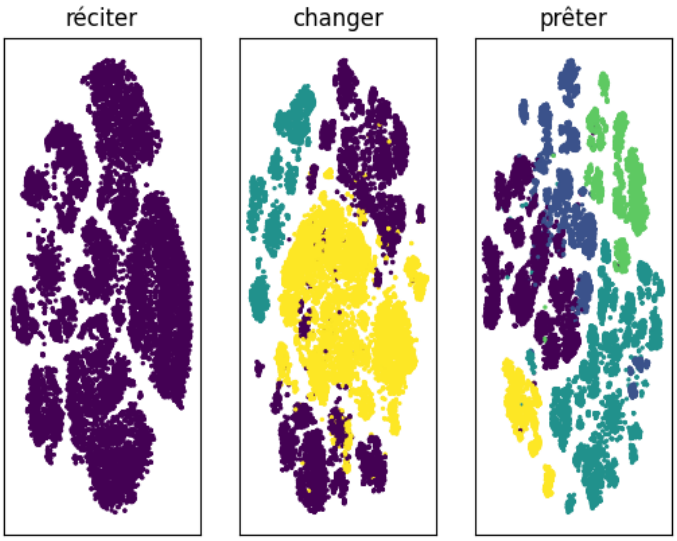

FIGURE 2.3 – Représentation en deux dimensions, par t-SNE, des plongements de mots contextuels obtenus avec le modèle de langue *CamemBERT* pour les verbes *réciter, changer et prêter*, classés en un, trois et cinq sens respectivement par l'algorithme de regroupement agglomératif avec un seuil de distance de 7 000.

Chapitre 3

Analyse syntaxique automatique

3.1 Notions élémentaires

Ce travail consiste à extraire les différents éléments syntaxiques imposés par les verbes les plus fréquents du français. Ces éléments syntaxiques ont plusieurs appellations selon la théorie utilisée ; Lucien Tesnière (1959) parle de *valence*, terme inspiré de la valence chimique, qui indique le nombre d'*actants* qu'un prédicat (verbe, nom ou adjectif) peut recevoir. Lucien Tesnière compare le verbe à « une sorte d'atome crochu susceptible d'exercer son attraction sur un nombre plus ou moins élevé d'actants, selon qu'il comporte un nombre plus ou moins élevé de crochets pour les maintenir dans sa dépendance » (Tesnière, 1959, p. 238). Noam Chomsky (1965), quant à lui, parle de *cadre de sous-catégorisation*, qui indique le nombre et la nature des *arguments* d'un prédicat. Ces différents termes étant utilisés pour décrire la même chose dans les travaux en TAL, nous allons, par souci de cohérence, parler ici de *valence* pour parler du comportement syntaxique des verbes en général, de *cadre syntaxique*, ou tout simplement de *cadre*, lorsque nous parlons d'un comportement syntaxique en particulier, et d'*arguments*.

Le cadre syntaxique d'un verbe indique son sujet et ses compléments essentiels (objet direct et objet indirect), par opposition aux compléments non essentiels (compléments de manière, de temps, de lieu, etc.) et aux éléments de négation (*ne... pas*). Un cadre syntaxique complet indique également les prépositions régies par le verbe. Les arguments syntaxiques correspondent souvent aux actants sémantiques des verbes, mais pas toujours. Par exemple, les pronoms impersonnels, notamment utilisés pour des verbes de météo comme dans *il pleut*, sont des sujets syntaxiques qui n'ont pas de fonction sémantique.

Parmi les arguments d'un verbe, certains sont dits obligatoires. Dans notre cas, cette appellation désignera les compléments qui doivent toujours se retrouver en présence du verbe. Par exemple, le verbe *prendre* a un complément direct obligatoire : *Il prend ses responsabilités* ne peut être réduit en **Il prend.* Il arrive que certains compléments obligatoires puissent s'élider : le verbe *manger*, par exemple, a un complément direct obligatoire qui est parfois implicite. Pour décrire la même situation, on peut tout aussi bien dire *Il mange son dîner* que *Il mange*. Enfin, certains compléments sont tout à fait optionnels ; ce sont généralement les compléments prépositionnels, qui sont favorisés par certains verbes, mais qui peuvent tout à fait être omis. Par exemple, le verbe *nager* est souvent accompagné d'un complément prépositionnel précédé de *dans*, comme dans *Je nage dans la mer.*

Le caractère optionnel ou obligatoire des compléments sera apparent via la fréquence des cadres syntaxiques que nous extrairons automatiquement : dans le cas du verbe *manger*, par exemple, nous pouvons, pour le même sens, nous attendre à avoir dans nos cadres syntaxiques principaux un cadre plus fréquent qui comporte un complément direct, et un autre cadre, moins fréquent, qui n'en comporte pas [1].

3.2 Universal Dependencies

Parmi les objectifs de ce travail, rappelons-le, figure celui de créer une méthode qui a le potentiel de s'appliquer sur plusieurs langues avec le moins d'ajustements possible. Le cadre d'annotation de UD [2] s'est imposé pour cette raison. Il s'agit d'un projet qui développe un cadre d'annotation qui peut s'appliquer au plus grand nombre de langues possible, c'est-à-dire qui annote de façon similaire des constructions similaires tout en permettant certaines extensions spécifiques au besoin. Ce projet a connu un développement impressionnant depuis la publication de la première version du guide d'annotation (Nivre *et al.*, 2016) ; aujourd'hui, près de 200 corpus arborés sont disponibles dans plus de 100 langues, et de nombreux analyseurs syntaxiques sont entraînés sur ces données et peuvent produire des annotations en UD.

Il existe certaines critiques par rapport à ce format (voir Osborne et Gerdes, 2019), notamment pour son traitement des mots-outils, et il continue d'être amélioré. Néanmoins, dans le cadre de ce travail, nous croyons qu'il est amplement suffisant pour identifier les éléments que nous souhaitons identifier, et sa couverture de nombreuses langues combinée à sa facilité

1. C'est effectivement ce que nous constatons, voir la ressource sur GitHub pour plus de détails.
2. `https://universaldependencies.org/`

d'utilisation via de nombreux outils en fait un format de choix. Nous présentons certains de ces outils ci-dessous.

3.3 Analyseurs syntaxiques automatiques

Les analyseurs syntaxiques suivants ne sont pas les seuls disponibles, et nous aurions pu en tester bien d'autres. Toutefois, ces outils sont utilisés fréquemment dans des tâches de TAL et sont réputés pour leurs bonnes performances.

3.3.1 *spaCy*

spaCy[3] est une librairie libre et gratuite pour le TAL en Python conçue pour un usage industriel, c'est-à-dire pour traiter une grande quantité de données. Elle peut effectuer de la reconnaissance d'entités nommées et une analyse en dépendance avec toutes les étapes préalables (tokenisation, lemmatisation, etc.) pour des dizaines de langues. Elle est conçue pour être facile à utiliser avec Python et fournit un outil de visualisation pour l'analyse syntaxique.

spaCy se démarque de ses compétiteurs par sa vitesse d'exécution : sur CPU, elle est environ 11 fois plus rapide que *Stanza* et 9 fois plus rapide que *UDPipe*.

3.3.2 *Stanza*

Stanza (Qi *et al.*, 2020) est une librairie Python pour le TAL. Elle offre les mêmes fonctionnalités d'analyse que *spaCy*, en plus de l'analyse en constituance, de l'analyse de sentiments et de l'identification de la langue. Comme *spaCy*, elle est disponible dans des dizaines de langues, et est très simple à utiliser.

3.3.3 *UDPipe*

UDPipe est une chaîne de traitement qui fournit l'analyse en dépendance de textes et toutes les étapes préalables. Des modèles préentraînés sont disponibles pour de nombreuses langues, mais si on dispose de données annotées, on peut également les utiliser pour entraîner *UDPipe* pour une nouvelle langue. Elle est disponible sous forme de librairie en Python mais aussi en Java, C++, Perl, etc.

3. `https://spacy.io/`

3.3.4 HOPS

HOPS (Grobol et Crabbé, 2021) est un analyseur syntaxique développé en 2021. Il vient avec plusieurs modèles de langues pré entraînés sur le français, mais il peut également être entraîné pour d'autres langues. Il peut être utilisé dans la ligne de commande ; dans ce cas, il faut donner en entrée un fichier au format CoNLL-U, ce qui implique une conversion des phrases dans ce format. Il peut également être utilisé comme composant dans une chaîne de traitement de *spaCy*. Nous n'avons pas testé la vitesse d'exécution de HOPS lorsqu'il fait partie d'une chaîne de traitement de *spaCy* ; toutefois, lorsqu'il est utilisé via la ligne de commande, il est significativement plus lent que les autres analyseurs.

3.4 Évaluation des analyseurs

Le tableau 3.1 indique les scores F_1 obtenus sur les corpus d'évaluation des corpus arborés pour le français de UD version 2.9 (Zeman *et al.*, 2021), sauf le corpus FTB, que nous n'avons pas considéré en raison de sa nature : il ne comporte que des mots isolés, tandis que cette recherche s'intéresse aux compléments des verbes et nécessite donc plutôt des phrases complètes. Les autres corpus ont été analysés par les quatre analyseurs syntaxiques énumérés en 3.3. Les résultats ont été évalués à l'aide du script d'évaluation CoNLL, version 2018 [4]. Nous avons considéré les mesures principales, soit :

- les caractéristiques universelles, de *universal features* en anglais (UFeats), qui incluent notamment le mode verbal, le nombre, le cas, etc. ;
- le score d'attachement non étiqueté, de *unlabeled attachment score* en anglais (UAS), soit le pourcentage des mots ayant le bon gouverneur ;
- le score d'attachement étiqueté, de *labeled attachment score* en anglais (LAS), soit le pourcentage des mots ayant le bon gouverneur et la bonne relation.

De ce tableau, nous pouvons tirer les conclusions suivantes :

- *Stanza* et HOPS obtiennent des résultats supérieurs pour tous les corpus et toutes les mesures considérées ;
- HOPS a le meilleur score moyen pour ce qui est du LAS, avec un score de 78,32. *Stanza*, quant à lui, a le meilleur score moyen pour ce qui est du UFeats (88,15) et du UAS (83,50) ;
- *Stanza* et HOPS ont des scores de UAS et LAS très similaires ; toutefois, *Stanza* a un

4. `https://github.com/CoNLL-UD-2018/UDPipe-Future/blob/master/conll18_ud_eval.py`

		UFeats	UAS	LAS
FQB	*Stanza*	76,92	**92,21**	85,62
	spaCy	**83,21**	82,49	62,83
	UDPipe	72,89	82,88	77,51
	HOPS	32,65	91,75	**87,09**
GSD	*Stanza*	**97,01**	**91,68**	**88,57**
	spaCy	77,40	74,49	65,90
	UDPipe	84,23	81,82	78,11
	HOPS	37,81	85,39	81,86
ParisStories	*Stanza*	**92,18**	**68,61**	**60,60**
	spaCy	75,38	60,36	47,38
	UDPipe	82,92	57,94	50,79
	HOPS	46,98	67,13	56,93
ParTUT	*Stanza*	**86,74**	86,05	82,08
	spaCy	79,66	77,98	69,93
	UDPipe	79,65	83,73	80,58
	HOPS	32,39	**88,61**	**85,53**
Rhapsodie	*Stanza*	**92,47**	**77,90**	**71,42**
	spaCy	77,07	68,50	55,45
	UDPipe	85,10	68,44	62,02
	HOPS	42,89	74,95	67,39
Sequoia	*Stanza*	83,57	84,53	81,11
	spaCy	**91,29**	82,24	75,50
	UDPipe	77,85	81,04	77,13
	HOPS	36,69	**92,34**	**91,10**
Moyenne	*Stanza*	**88,15**	**83,50**	78,23
	spaCy	80,66	74,34	62,83
	UDPipe	80,44	75,98	71,02
	HOPS	36,24	83,36	**78,32**

TABLEAU 3.1 – Résultats des analyseurs *Stanza*, *spaCy*, *UDPipe* et HOPS sur les différents corpus de test de UD (en F_1). À noter que les résultats indiqués peuvent différer des résultats affichés sur les sites web des différents modèles en raison des versions des analyseurs utilisés et des corpus UD.

score significativement supérieur à HOPS pour le UFeats, mesure importante pour notre travail puisque c'est là où sont notées les informations de mode des verbes.

- *Stanza* était plus facile à exécuter que HOPS au moment de faire nos tests, puisqu'il bénéficiait d'une API Python (HOPS peut maintenant être intégré dans une chaîne de traitement de *spaCy*, ce qui facilite grandement son utilisation), tandis que nous n'avons

pu exécuter HOPS que dans un environnement Linux et à partir de la ligne de commande.

En conclusion, ***Stanza*** était le meilleur analyseur syntaxique parmi ceux que nous avons testés lorsque nous prenions en compte ses scores (83,29 % en moyenne) et sa facilité d'exécution. C'est celui que nous avons utilisé pour le reste de nos expériences.

3.5 Méthode d'extraction des cadres syntaxiques

Pour extraire les cadres syntaxiques, nous considérons séparément chacun des sens des verbes déterminés dans le chapitre 2. Chacune des phrases contenant le verbe associé au sens est analysée avec *Stanza*. Cette analyse nous permet d'identifier lesquels de ces verbes sont dans les clauses principales : ce sont seulement ces verbes que nous préservons, puisque ces verbes ont généralement des cadres syntaxiques plus complets. Nous ne considérons également que les verbes finis et dont le mode n'est pas l'impératif.

Par la suite, nous distinguons les arguments verbaux des autres types d'arguments en ignorant ceux dont la relation est `dislocated`, `advmod`, `obl:mod`, `advcl`, `punct`, `parataxis`, `conj`, `amod`, `det`, `nmod`, `case`, `cc`, `aux`, `aux:tense`, `aux:pass` et `mark` (voir la documentation de UD [5] pour les définitions de ces relations syntaxiques). Si les compléments trouvés gouvernent une préposition, alors cette préposition sera également conservée.

Parmi les compléments résultants, nous distinguons les arguments réels des ajouts. Nous considérons les dépendants régis par les relations `nsubj`, `obj`, `iobj`, `csubj`, `ccomp`, `xcomp` et `expl` comme des arguments réels, et les dépendants régis par les relations `obl`, `vocative` et `dep:comp` comme des ajouts. Les cadres contenant uniquement des arguments réels seront considérés comme des cadres "de base" et ceux contenant des ajouts seront considérés comme des cadres "optionnels".

3.6 Filtrage

Nos données comporteront nécessairement une quantité importante de bruit, dû aux inévitables erreurs engendrées par les outils automatiques, d'autant plus considéré le nombre d'étapes dans notre chaîne de traitement. Il peut y en avoir dès l'extraction des données, puisque Sketch Engine fait sa propre lemmatisation et peut mal étiqueter certains mots [6] ; l'étape d'induction de sens a une fiabilité plutôt basse (voir 2.3.2) ; et les analyseurs syntaxiques

5. `https://universaldependencies.org/u/dep/all.html`
6. Par exemple, pour le verbe *boiter*, figurent des phrases comme *Livraison vrac boite de 12* ou *Toyota Land Cruiser KDJ95 boite auto, équipé / renforcé raid*)

comportent également leur lot d'erreurs (voir tableau 3.1). Il est donc particulièrement important de faire un filtrage sur nos données pour éliminer ces erreurs le plus possible. Nous ferons ce filtrage sur le résultat final, c'est-à-dire sur les cadres syntaxiques que nous avons extraits automatiquement. Pour cela, nous devons déterminer une mesure d'association adéquate, puis l'appliquer sur nos données.

3.6.1 Mesure d'association

Pour le filtrage de cadres syntaxiques, nous souhaitons déterminer s'il existe une réelle association entre un cadre syntaxique donné et le verbe cible. Pour déterminer la meilleure mesure à utiliser, nous nous sommes basés sur Korhonen (2002). Dans sa thèse, elle a comparé trois différents tests d'hypothèse : le test binomial, le test du rapport de vraisemblance, et le test de Student (test t). Ces trois méthodes ont donné des résultats peu satisfaisants, ce qui a incité l'auteure à se tourner vers une simple méthode d'estimation de vraisemblance maximale, qui fait complètement fi de la notion de pertinence.

Malgré ce que peut faire croire son nom, cette méthode ne demande à peu près aucun calcul. Elle demande simplement de compter le nombre d'occurrences d'un cadre syntaxique avec le verbe donné (un *sens* du verbe donné dans notre cas), et de le diviser par le nombre d'occurrences total du verbe (ou du sens du verbe). Le cadre qui donne le quotient le plus élevé est considéré comme le cadre le plus fréquent du verbe. Cette méthode donne des résultats largement supérieurs aux autres méthodes testées, comme l'ont également démontré Simon *et al.* (2010) et Rambelli *et al.* (2016). C'est aussi la méthode qui a été utilisée par Messiant *et al.* (2008). La formule correspondante est illustrée en 3.1. Nous ne gardons que les cadres pour lesquels le résultat est supérieur à une valeur minimale, que nous déterminons plus loin.

$$\frac{\text{Nombre d'occurrences du cadre}_i}{\text{Nombre d'occurrences du verbe}_j \text{ dans son sens}_x} \tag{3.1}$$

3.6.2 Méthode

Pour déterminer la fréquence relative minimale d'un cadre syntaxique pour qu'il soit considéré comme correct, nous avons décidé de comparer nos cadres avec ceux de *Dicovalence*. *Dicovalence* a l'avantage d'être une ressource faite manuellement, et donc d'assez bonne qualité, en plus d'être facile à consulter à la fois pour les humains et pour la machine. En effet, pour chaque cadre syntaxique, ou "usage", un exemple est donné du verbe en contexte, nous permettant donc également d'avoir une idée du sens de ce verbe.

Toutefois, *Dicovalence* ne fait pas spécifiquement de distinction de sens entre les verbes ; ainsi, il peut arriver qu'un seul cadre syntaxique corresponde à un sens, mais souvent un sens sera représenté par plusieurs cadres syntaxiques. Le verbe *rapprocher*, par exemple, a six entrées dans le *Dicovalence*, avec une distinction entre *rapprocher* et *se rapprocher* (voir annexe A). Le lecteur pourra juger du nombre de sens exprimés dans les six entrées de ce verbe. Pour évaluer nos cadres syntaxiques, nous ignorons donc les distinctions de sens, et observons plutôt le nombre de cadres syntaxiques différents pour un même verbe indiqués dans le *Dicovalence* pour les comparer avec les différents cadres syntaxiques extraits dans nos données.

22 verbes ont été sélectionnés au hasard dans nos données. Parmi ces 22 verbes, deux n'étaient pas recensés dans le *Dicovalence*, soit *cuber* et *mémoriser*. En fin de compte, 20 verbes ont été utilisés pour déterminer la fréquence relative minimale pour qu'un cadre syntaxique soit considéré comme tel : *camper, échanger, marquer, accroître, inverser, voyager, congeler, démunir, garder, rénover, dépêcher, sortir, reconnaître, imiter, assortir, expérimenter, recouvrir, loger, explorer* et *vérifier*.

Nous avons déterminé ce seuil de façon manuelle en suivant les étapes suivantes pour chacun des 20 verbes :

1. Nous avons extrait les cadres syntaxiques du verbe dans *Dicovalence*.
2. Nous avons extrait les cadres syntaxiques accompagnés de leur fréquence relative pour chaque sens du verbe trouvé par notre algorithme.
3. Parmi ces derniers, nous différencions les cadres "de base" qui contiennent uniquement des arguments obligatoires comme les sujets et les objets, des compléments facultatifs comme les compléments prépositionnels. Dans certains cas également, des compléments obligatoires peuvent subir une élision. Nous tentons donc d'établir un seuil minimal pour chacune de ces catégories de compléments. Dans nos données, ces trois types de compléments peuvent être séparés : par exemple, nous pouvons trouver un cadre contenant seulement un sujet, un autre contenant un sujet et un complément prépositionnel introduit par la préposition *à*, et un autre contenant un sujet, un complément direct et un complément prépositionnel également introduit par la préposition *à*.
4. Nous tentons d'établir un seuil qui maximise à la fois la précision et le rappel, c'est-à-dire qui couvre le plus de compléments indiqués par *Dicovalence* possible, tout en minimisant le nombre de cadres syntaxiques erronés.

À noter qu'en raison du format UD, les attributs, comme *beau* dans *il est beau*, ne seront pas capturés. UD annote les adjectifs eux-mêmes comme les gouverneurs des verbes d'état,

tandis que nous ne sélectionnons que les racines verbales des phrases.

À noter également que dans UD, les relations syntaxiques expriment également la partie du discours du dépendant. Il faut donc, par exemple, différencier les sujets nominaux (`nsubj`) des sujets verbaux (`csubj`), qui sont parfois amalgamés dans *Dicovalence*.

3.7 Résultats

Les seuils que nous avons choisis à la suite de notre examen manuel sont :

- 14 % pour les cadres syntaxiques de base, c'est-à-dire qui ne contiennent que des arguments obligatoires ;
- 9 % pour les cadres syntaxiques contenant un complément obligatoire qui peut s'élider ;
- 4 % pour les cadres syntaxiques contenant un complément optionnel.

Pour vérifier ces seuils, nous avons comparé nos résultats avec ceux de *Dicovalence* pour 20 autres verbes, choisis au hasard encore une fois : *recréer, adopter, blanchir, relater, surmonter, chérir, délaisser, distribuer, déléguer, préoccuper, concéder, annuler, mêler, décorer, libérer, suivre, revendiquer, allier, balayer* et *effrayer*. Nous avons comparé les cadres syntaxiques de chacun des verbes de *Dicovalence* et ceux que nous avons extraits et filtrés. Le rappel était calculé selon la formule 3.2, et la précision était calculée en fonction de la formule 3.3. Le détail de chacun des cadres syntaxiques peut être consulté dans le tableau 3.2. Les cadres syntaxiques de Dicovalence spécifient pour chaque position sa fonction syntaxique (`subj`, `obj`, etc.), son caractère obligatoire ou facultatif (indiqué par un point d'interrogation dans le *Dicovalence*), ses réalisations syntagmatiques possibles (pronom, nom, etc.) et certaines restrictions de sélections. Ici, les cadres syntaxiques contenant un complément facultatif sont divisés en deux cadres différents : un contenant le complément facultatif, l'autre ne le contenant pas. Cela nous permet de comparer ces cadres directement avec nos données, qui sont divisées de cette façon. Les cadres syntaxiques équivalents sont placés sur la même ligne dans notre tableau.

$$\text{Rappel} = \frac{\text{Nombre de cadres syntaxiques communs}}{\text{Nombre de cadres syntaxiques dans Dicovalence}} \tag{3.2}$$

$$\text{Précision} = \frac{\text{Nombre de cadres syntaxiques communs}}{\text{Nombre de cadres syntaxiques extraits dans nos données}} \tag{3.3}$$

Verbe	***Dicovalence***	**Nos données**	**P**	**R**
recréer	1) `subj:pron|n, obj:pron|n`	1) `nsubj, obj`	1/1	1/1

Verbe	*Dicovalence*	Nos données	P	R
adopter	1) subj:pron|n, obj:pron|n	1) nsubj, obj	1/1	1/1
blanchir	1) subj:pron|n, obj:pron|n	1) nsubj, obj	2/2	2/3
	2) subj:pron|n, obj:pron|n,objde:pron|n	2) nsubj		
	3) subj:pron|n			
relater	1) subj:pron|n, obj:pron|n	1) nsubj, obj	1/1	1/2
	2) subj:pron|n, obj:compl|inf|indirq			
surmonter	1) subj:pron|n, obj:pron|n	1) nsubj, obj	1/1	1/1
chérir	1) subj:pron|n, obj:pron|n	1) nsubj, obj	1/5	1/1
		2) obj		
		3) nsubj		
		4) nsubj, nsubj, obj		
		5) nsubj, nsubj		
délaisser	1) subj:pron|n, obj:pron|n	1) nsubj, obj	1/1	1/3
	2) subj:pron|n, obj:pron|n, objp<pour>:pron|n			
	3) subj:pron|n, obj:pron|n, objp<pour>:inf			
distribuer	1) subj:pron|n, obj:pron|n, objà:pron|n	1) nsubj, obj, obl:arg à	2/2	2/5
	2) subj:pron|n, obj:pron|n	2) nsubj, obj		
	3) subj:pron|n, obj:pron|n, loc<>:pron|n			
	4) pseudo_se, subj:pron|n			
	5) pseudo_se, subj:pron|n, loc<>pron|n			
déléguer	1) subj:pron|n, obj:pron|n, objà:pron|n	1) nsubj, obj, obl:arg à	1/2	1/1
		2) nsubj, obj		
préoccuper	1) subj:pron|n, obj:pron|n	1) nsubj, obj	2/4	2/5
	2) subj:compl, obj:pron|n			
	3) subj:de_inf, obj:pron|n			
	4) pseudo_se, subj:pron|n, objde:pron|n	2) dep:comp, nsubj, obl:arg-de		
	5) pseudo_se, subj:pron|n, objde:de_inf			
		3) dep:comp, nsubj, obj		
		4) nsubj, obj, obl:arg-de		
concéder	1) subj:pron|n, obj:pron|n	1) nsubj, obj	3/3	3/4
	2) subj:pron|n, obj:pron|n, obj à	2) nsubj, obj, obl:arg à		
	3) subj:pron|n, obj:compl	3) nsubj, ccomp que		
	4) subj:pron|n, obj:compl, objà:pron|n			
annuler	1) subj:pron|n, obj:pron|n	1) nsubj, obj	1/2	1/1
		2) expl:pass, nsubj:pass		
mêler	1) subj:pron|n, obj:pron|n, objp <avec>:pron|n	1) nsubj, obj	2/2	2/8
	2) subj:pron|n, obj:pron|n	2) nsubj, obj, obl:arg à		
	3) subj:pron|n, obj:pron|n, objà:pron|n			
	4) subj:pron|n, obj:pron|n, objde:pron|n			
	5) pseudo_se, subj:pron|n, objà:pron|n			
	6) pseudo_se, subj:pron|n, objde:pron|n			
	7) pseudo_se, subj:pron|n, objde:de_inf			
	8) pseudo_se, subj:pron|n			
décorer	1) subj:pron|n, obj:pron|n	1) nsubj, obj	1/1	1/3
	2) subj:pron|n, obj:pron|n, objde:pron|n			
	3) subj:pron|n			
libérer	1) subj:pron|n, obj:pron|n	1) nsubj, obj	2/3	2/3
	2) subj:pron|n, obj:pron|n, objde:pron|n	2) nsubj, obj, obl:arg de		
	3) pseudo_se, subj:pron|n			
		3) nsubj		

Verbe	*Dicovalence*	Nos données	P	R
suivre	1) `subj:pron|n, obj:pron|n`	1) `nsubj, obj`	3/3	3/3
	2) `subj:pron|n, obj:pron|n, loc<>:pron|n`	2) `nsubj, obl:arg à|dans`		
	3) `subj:pron|n`	3) `nsubj`		
revendiquer	1) `subj:pron|n, obj:pron|n`	1) `nsubj, obj`	1/1	1/1
allier	1) `subj:pron|n, obj:pron|n`	1) `nsubj, obj`	5/5	5/6
	2) `subj:pron|n, obj:pron|n, objp<avec>:pron|n`			
	3) `subj:pron|n, obj:pron|n, objp<à>:pron|n`	2) `nsubj, obj, obl:arg à`		
	4) `pseudo_se, subj:pron|n`	3) `dep:comp, nsubj`		
	5) `pseudo_se, subj:pron|n, objp<avec>:pron|n`	4) `dep:comp, nsubj, obl:arg avec`		
	6) `pseudo_se, subj:pron|n, objp<à>:pron|n`	5) `dep:comp, nsubj, obl:arg à`		
balayer	1) `subj:pron|n`	1) `nsubj`	2/3	2/2
	2) `subj:pron|n, obj:pron|n`	2) `nsubj, obj`		
		3) `nsubj, obj, obl:arg de`		
effrayer	1) `subj:pron|n, obj:pron|n`	1) `nsubj, obj`	1/2	1/5
	2) `subj:compl, obj:pron|n`			
	3) `subj:de_inf, obj:pron|n`			
	4) `pseudo_se, subj:pron|n`			
	5) `pseudo_se, subj:pron|n, objde:pron|n`			
		2) `nsubj, nsubj`		
			34/46	34/59
			73,91 %	**57,62 %**
			F_1 =	**64,76 %**

TABLEAU 3.2 – Comparaison entre les cadres syntaxiques trouvés dans *Dicovalence* et ceux dans nos données, avec les scores de précision (P) et de rappel (R).

Nous pouvons constater qu'en moyenne, la précision est bien meilleure que le rappel (73,91 % contre 57,62 %). Ceci peut s'expliquer par plusieurs facteurs :

- **Exhaustivité** : *Dicovalence* couvre le plus d'usages syntaxiques possible, y compris les moins fréquents. Quant à nous, nous appliquons un filtre sévère sur nos résultats afin d'éliminer le plus de bruit possible, ce qui peut entraîner une couverture diminuée au profit de données plus correctes.
- **Compléments optionnels** : Ensuite, *Dicovalence* peut inclure des compléments optionnels qui ne sont pas identifiés comme tels (c'est-à-dire sans "?" devant), par exemple, les `objp`, qui équivalent aux compléments prépositionnels de la grammaire traditionnelle. Les auteurs de *Dicovalence* précisent d'ailleurs que ce type de complément ne se cliticise pas, ce qui indique qu'il n'est pas essentiel. Quant à nous, nous ne considérons que les compléments qui sont identifiés comme des arguments réels par *Stanza*, ce qui peut faire en sorte que les cadres syntaxiques qui en contiennent sont ignorés.
- **Pronoms réflexifs** : Enfin, plusieurs cadres syntaxiques qui incluaient le pronom *se* ne se retrouvent pas dans nos données. Étant donné que la prise en charge de ce type de

pronom par UD n'est pas consistante, il est possible que la proportion totale de ces cas soit divisée en deux, et que leurs plus petits pourcentages fassent qu'ils disparaissent lors du filtrage.

Même si nous n'avons pas changé les cadres syntaxiques eux-mêmes (nous n'avons pas, notamment, ignoré des compléments dans *Dicovalence* qui auraient pu être jugés optionnels), quelques mises au point ont été faites pour pouvoir adéquatement comparer les cadres de *Dicovalence* avec ceux que nous avons extraits de nos données.

1. **Un cadre dans *Dicovalence* peut correspondre à plusieurs cadres UD.** *Dicovalence* propose des cadres avec trois éléments principaux : les sujets (`subj`), les compléments (`obj`), et les pronoms réflexifs *se* (`pseudo_se`). Les sujets et les objets sont ensuite accompagnés d'une indication de la partie du discours requise. Dans UD, les parties du discours sont implicites dans les étiquettes de relations : par exemple, `nsubj` indique un sujet nominal (pronom ou nom), tandis que `csubj` indique un sujet verbal. De la même façon, en UD, un complément nominal sera étiqueté `obj`, tandis qu'un complément verbal sera étiqueté `ccomp` ou `xcomp`. Il arrive également qu'un complément `loc` apparaisse dans *Dicovalence* : dans ce cas, si les propositions que nous trouvons dans nos cadres peuvent servir à indiquer un lieu, nous considérons qu'ils sont équivalents.
2. **Un cadre syntaxique contenant un élément optionnel sera divisé en deux.** Par exemple, nous avons divisé un cadre comme `subj:pron|n`, `obj:pron|n`, `?objp<pour>:pron|n` (où le point d'interrogation indique un complément optionnel) en deux, le premier étant `subj:pron|n`, `obj:pron|n` et le deuxième étant `subj:pron|n`, `obj:pron|n`, `objp<pour>:pron|n`, car ce sont ces deux cadres qui sont susceptibles d'apparaître dans nos données.
3. **Les relations UD implicitent souvent la partie du discours du dépendant, mais pas toujours.** La relation `xcomp`, par exemple, peut désigner soit un complément verbal, soit un complément adjectival.
4. **Les compléments indirects.** La relation `iobj` en UD est dédiée aux compléments indirects. Toutefois, dans nos données, nous ne trouvons que la relation `iobj:agent`, qui concerne les agents des verbes à contrôle. Les compléments indirects sont plutôt désignés par la relation `obl:arg`, pour les "arguments obliques", par opposition aux ajouts, désignés par la relation simple `obl`, comme indiqué dans le guide d'annotation en ligne [7].

Le tableau 3.3 présente la correspondance entre les arguments qui figuraient dans les

7. `https://universaldependencies.org/u/dep/all.html#al-u-dep/obl`

entrées que nous avons observées dans *Dicovalence* et ceux fournis par *Stanza*, c'est-à-dire par le cadre d'annotation UD.

Dicovalence	**Stanza**
subj:pron|n	nsubj
obj:pron|n	obj
objde|à:pron|n	obl:arg de|à
obj:compl|inf|indirq	ccomp|xcomp
pseudo_se	dep:comp|expl:pass
loc<>pron|n	obl:arg dans|à
objp<avec|pour|à|...>:pron|n	obl:arg avec|pour|à...
subj:de_inf	csubj de
objp<pour>:inf	ccomp|xcomp pour
subj:compl	csubj
objde:de_inf	ccomp|xcomp de

TABLEAU 3.3 – Correspondance entre les arguments indiqués dans *Dicovalence* et les relations de UD.

3.8 Conclusion

En conclusion, les seuils de filtrage que nous avons déterminés nous donnent des résultats que nous considérons satisfaisants pour notre travail. En effet, la précision est plutôt bonne (73,91 %), et le rappel inférieur (57,62 %) peut s'expliquer par différents facteurs, comme l'exhaustivité des cadres couverts et la présence de compléments optionnels dans *Dicovalence*, ainsi que le traitement inconsistant des pronoms réflexifs par *Stanza*. Ces facteurs indiquent notamment que *Dicovalence* n'est pas nécessairement la meilleure ressource sur laquelle nous évaluer. Dans les travaux futurs, il serait intéressant de sélectionner un dictionnaire de référence plus proche de nos besoins. De plus, l'évaluation des cadres syntaxiques a été faite manuellement, c'est-à-dire que la correspondance entre les arguments de *Dicovalence* et ceux de *Stanza* n'a pas été automatisée. Une telle automatisation de l'évaluation serait pertinente à réaliser à l'avenir.

Chapitre 4

Résultats

4.1 Présentation du dictionnaire

Notre dictionnaire de valence comporte 1 950 entrées, chacune correspondant à un lemme verbal (ou lemme identifié comme un verbe par Sketch Engine). En moyenne, chacune de ces entrées comporte 2,23 sens. La distribution de ces sens est illustrée à la figure 4.1.

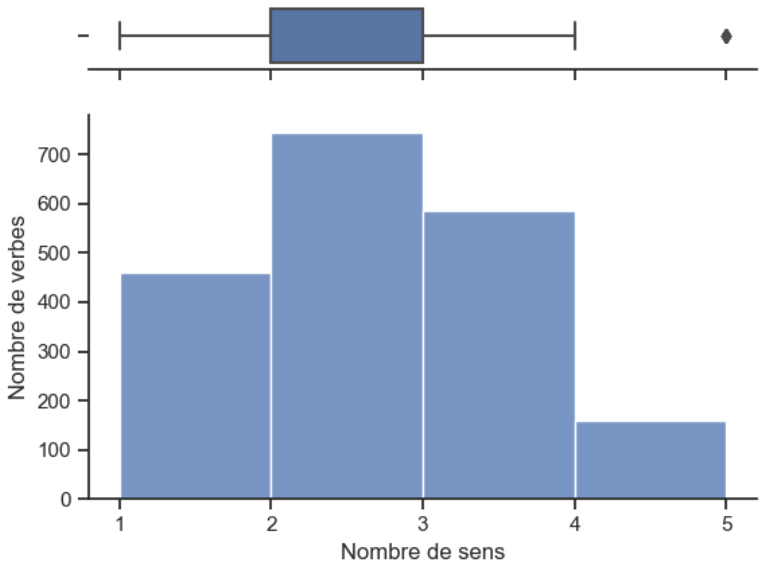

FIGURE 4.1 – Distribution du nombre de sens par verbe dans notre dictionnaire

Le nombre d'entrées dans notre dictionnaire est inférieur aux 1958 lemmes extraits au chapitre 2 ; cela est dû au filtrage effectué lors de l'extraction des cadres syntaxiques (3.6).

Si un verbe ne compte aucun cadre syntaxique qui atteint ou dépasse le seuil de 14 % d'occurrence dans nos données, alors l'entrée de ce verbe est nécessairement vide. Cela peut se produire quand les phrases d'exemples sont trop disparates, ce qui est généralement signe d'erreurs d'analyse syntaxique ou d'une qualité d'exemples inférieure. Les verbes qui ont été ignorés de cette façon sont *étonner, trailer, dépourvoir, entrainer, téléviser, usager, parfaire* et *issir*. Le nombre de sens, toutefois, est le même que celui que nous avions obtenu dans l'étape d'induction de sens.

4.2 Structure générale des entrées

Pour comprendre la structure de notre dictionnaire, considérons l'entrée du verbe *nager*.

Listing 4.1 – Entrée du verbe *nager* de notre dictionnaire

```
"nager": {
  "sens": [
   [{"nom": "nsubj",
     "freq": 0.69,
     "exemples": [
      "Quelqu'un nageait à proximité?",
      "J'ai tout de suite compris qu'ils étaient là pour moi. une heure
         après ils nageaient dans mon aquarium.",
      "La planète Malaisus est composée d'eau; un monstre similaire à
         celui qui sert de monture au joueur y nage au milieu de petits
         poissons.",
      "Je nageais dans une poche de liquide amniotique d'un bleu
         laiteux, mes membres maladroits agités de soubresauts, je
         poussais sur ces membranes avec des gestes vains, la tête hors
         du sac mais aveuglée par la guirlande du cordon ombilical.",
      "Billot nageait dans la joie."],
       "sous-cadres": [
           {"nom": "nsubj-obl:arg|dans",
           "freq": 0.08,
           "exemples": [
               "C'est où je nageais dans l'eau, où j'ai des souvenirs
                  extraordinaires.",
               "Un groupe d'eiders nagent dans le fjord, puis un groupe
                  de bernaches.",
```

```
                "Cela pourrait amener son lot d'incohérences, comme le
                   fait que certaines guildes ne se seraient jamais
                   côtoyées RP parlant par exemple. (Là, on nagerait
                   dans le méta)Je trouve que les systèmes actuels
                   (rumeurs et autres posts de présentation/recrutement
                   RP) remplissent déjà très bien le rôle de
                   rapprochement.",
                "Trois familles de canards nagent dans la mare, chacune
                   composée de Maman Cane et de ses canetons.",
                "On nageait dans le sordide, pire même, on y buvait la
                   tasse."]}]}
 ],
[{"nom": "nsubj-obj",
  "freq": 1.0,
  "exemples": [
   "Resultats Finale 100 m nage libre Champ.",
   "Resultats demies 100 m nage libre Champ."],
     "sous-cadres": []}
  ],
[{"nom": "nsubj-obj-xcomp",
  "freq": 0.14,
  "exemples": [
   "Résultat de la finale du relais 4 x 50 m nage libre des
      championnats d'Europe en petit bassin 2010, sur
      swimrankings.net."],
     "sous-cadres": []},
 {"nom": "nsubj-obj",
  "freq": 0.86,
  "exemples": [
   "le proble8me que posent ces combinaisons, la finale du 200 m nage
      libre hommes vient de porter une de9monstration
      e9clatante.Biedermann, M. muscles, l a emporte9 sur Phelps.",
   "L'épreuve de 100 m nage libre hommes des Jeux olympiques d'été de
      2012 a eu lieu le 31 juillet et le 1er août au London Aquatics
      Centre[1].",
   "L'épreuve de 200 m nage libre hommes des Jeux olympiques d'été de
      2012 a lieu le 29 et le 30 juillet au London Aquatics
      Centre[1]."],
     "sous-cadres": []}
 ],
[{"nom": "nsubj",
  "freq": 0.74,
```

```
 "exemples": [
  "Fête de l'eau – Ma Guyane nage Jeudi 15 décembre à 10h à l'école
     Abriba de Matoury Le dispositif ma Guyane nage a pour but
     d'apprendre à nager aux enfants et aux adultes dans les
     quartiers ou les communes qui ne disposent pas de bassin
     d'apprentissage de la natation.",
  "Ici on nage en plein militantisme décomplexé. --RH
     2A00:5E80:109F:2E00:3EDF:BDFF:FEA0:E61C (discuter) 2 mars 2014 à
      11:06 (CET)",
  "Et sur le plan du vocabulaire (fiction, non-fiction, on nage dans
     les anglicismes de traduction). --Elnon (discuter) 12 février
     2015 à 12:36 (CET)",
  "La Cigale des mers nage dans le succès",
  "Elle nage et se déplace sur terre indifféremment."],
    "sous-cadres": [
        {"nom": "nsubj-obl:arg|dans",
        "freq": 0.07,
        "exemples": [
            "Ou encore la brasse, je nage dans le texte.",
            "Le hareng nage dans les eaux de l'Atlantique Nord où il
               suit le plancton.",
            "Du désert, on nage dans une twilight zone, flottant
               entre l'espace et les terrains pétrolifères, un
               territoire large que notre imagination s'appropriera
               encore une fois.",
            "Ignorant les secrets de la Topologie À l'espace
               infligée, et toi qui l'étudies, Il nage dans l'erreur
               où son langage est pris.",
            "Le couple nage actuellement dans le bonheur et le
               rappeur a tenu à en parler un peu plus à la radio
               lors d'une interview."]}]},
{"nom": "nsubj-obj",
 "freq": 0.09,
 "exemples": [
  "Elle nage , la grotteElle nage, la grotte, avec moi dedans, dans
     toute la maison.",
  "L'année suivante, en 1925, elle nage 21 milles (33,8 km) en
     parcourant la baie de New York, de Manhattan à Sandy Hook, ce
     qui lui prend sept heures.",
  "Il nage plus de 3 000 km par saison.",
  "Weber-Gale nage les séries éliminatoires mais pas la finale dans
     laquelle le relais américain est sacré champion du monde.",
```

```
      "Elle nage , marche, ..."],
        "sous-cadres": []}]]},
```

Les différents champs sont les suivants :

- Le champ “sens” contient une ou plusieurs listes, chacune correspondant à un sens du verbe, et contenant le ou les différents cadres associés à ce sens. Par exemple, le verbe *nager* comporte quatre sens.
- Les champs “nom” contiennent chacun un cadre syntaxique obligatoire. Ces cadres syntaxiques sont ceux qui ont une fréquence relative de 14 % ou plus. Si ces cadres comportent des variantes où un argument peut être omis, et cette variante a une fréquence relative de 9 % ou plus, alors nous trouverons également ces variantes parmi les cadres principaux. C’est le cas des compléments qui sont obligatoires, mais qui peuvent s’élider, comme *manger* : ce verbe a un objet direct obligatoire, mais qui peut être omis dans certains cas (par exemple, on peut tout aussi bien dire *Je mange* que *Je mange un plat*, sans changement significatif de sens). Par exemple, le cadre du premier sens de *nager* est `nsubj`.
- La fréquence relative est indiquée dans le champ “freq”. Cette fréquence a une valeur maximale de 1. Par exemple, la fréquence du cadre du premier sens de `nager` est 0,69, ou 69 %.
- Les cinq exemples (ou moins) de ce cadre en contexte sont indiqués sous le champ “exemples”. Ces exemples servent à identifier les sens des verbes, puisque ceux-ci ne correspondent pas à des sens répertoriés dans une base de données. Ils ont tous été sélectionnés au hasard. [1] Si nous trouvons moins que cinq exemples, c’est un signe qu’il y avait très peu de phrases ayant été classées comme appartenant à ce sens.
- Pour chaque cadre, nous pouvons trouver, s’il y a lieu, un ou plusieurs sous-cadres syntaxiques, qui sont indiqués sous le champ “sous-cadres”. Ces sous-cadres sont ceux qui contiennent des arguments optionnels, comme les compléments prépositionnels. Ceux-ci doivent avoir une fréquence relative d’au moins 4 % et être associés à un des cadres principaux du sens du verbe. La structure de leur entrée est la même que celle des cadres principaux. Par exemple, le cadre du premier sens du verbe *nager* comporte un sous-cadre : `nsubj-obl:arg|dans`, qui apparaît dans 8 % des cas pour ce sens.

Regardons maintenant l’entrée de ce verbe plus en détail. Quatre sens ont été distingués pour le verbe *nager*. Le premier comporte un cadre syntaxique principal, avec une fréquence

1. Attention aux âmes sensibles : nous n’avons pas du tout contrôlé le type d’exemples présents dans le dictionnaire, et ces exemples sont tirés du web. Les sources sont parfois très douteuses, et les exemples pourraient choquer certaines personnes.

de 69 % : `nsubj`. Comme exemples de ce cadre, nous avons :

1. *Quelqu'un nageait à proximité ?*
2. *J'ai tout de suite compris qu'il était là pour moi. une heure après ils nageait dans mon aquarium.*
3. *La planète Malaisus est composée d'eau ; un monstre similaire à celui qui sert de monture au joueur y nage au milieu de petits poissons.*
4. *Je nageais dans une poche de liquide amniotique . . .*
5. *Billot nageait dans la joie.*

Ce cadre syntaxique principal possède également une variation avec un complément prépositionnel précédé de *dans*, qui survient dans 8 % des cas. Les exemples donnés sont les suivants :

1. *C'est où je nageais dans l'eau. . .*
2. *Un groupe d'eiders nagent dans le fjord. . .*
3. *. . . On nageait dans le meta. . .*
4. *Trois familles de canards nagent dans la mare. . .*
5. *On nageait dans le sordide. . .*

Ces exemples nous indiquent que ce sens de *nager* correspond au sens littéral de 'se mouvoir dans l'eau'. Seuls le dernier exemple du cadre principal et les troisième et cinquième exemples du sous-cadre contrarient cette interprétation : on peut donc supposer que ces exemples ont été mal classés.

Les deuxième et troisième sens présentent tous les deux moins que cinq exemples. Ceci est un indicateur que ces sens étaient composés d'un très petit échantillon de phrases, et en effet, ils ont comme origine une erreur d'étiquetage de partie du discours ; le nom *nage* a été confondu avec le verbe *nager*. C'est un cas particulier où à la fois Sketch Engine et *Stanza* ont mal identifié la partie du discours du token. Comme exemples de phrases, nous avons notamment :

- *Resultats Finale 1000 m nage libre Champ.* (2 fois)
- *Résultat de la finale du relais 4 x 50 m nage libre. . .*

Le quatrième sens a, comme le premier, `nsubj` comme seul cadre syntaxique principal, qui survient dans 74 % des cas. Les exemples donnés sont :

1. *Ma Guyane nage. . .*
2. *On nage en plein militantisme décomplexé. . .*

3. *On nage dans les anglicismes de traduction…*
4. *La Cigale des mers dans le succès.*
5. *Elle nage et se déplace sur terre indifféremment.*

Encore une fois, ce cadre syntaxique peut survenir avec un complément prépositionnel précédé de *dans*, dans 7 % des cas :

1. *Ou encore la brasse, je nage dans le texte.*
2. *Le hareng nage dans les eaux de l'Atlantique Nord…*
3. *Du désert, on nage dans une twilight zone…*
4. *Il nage dans l'erreur…*
5. *Le couple nage actuellement dans le bonheur…*

Ces exemples incitent à interpréter ce sens comme métaphorique, signifiant quelque chose comme 'être envahi d'un sentiment'. À noter qu'ils comportent tous un complément de lieu, y compris ceux qui figurent uniquement dans le cadre principal, ce qui indique que l'analyseur syntaxique n'a pas réussi à différencier correctement les compléments indirects des ajouts.

Enfin, un cadre `nsubj, obj` qui appartient également à ce sens métaphorique est composé de phrases qui ont visiblement été mal classées, qui font toutes penser au sens littéral. Ce cadre survient dans 9 % des cas, ce qui laisse pense que le complément d'objet est élidé la plupart du temps.

Les figures 4.2 à 4.6 présentent quelques cadres syntaxiques, dans leur contexte et isolés, dans des exemples tirés de nos données.

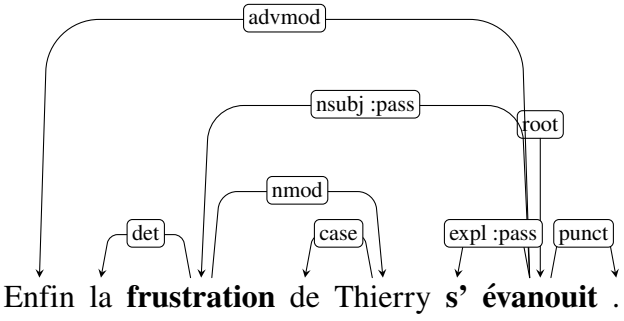

FIGURE 4.2 – Représentation du cadre `nsubj:pass-expl:pass` dans son contexte

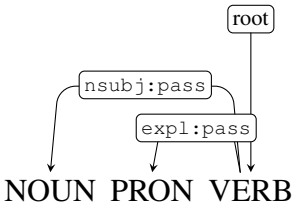

FIGURE 4.3 – Représentation du cadre `nsubj:pass-expl:pass` avec ses parties du discours

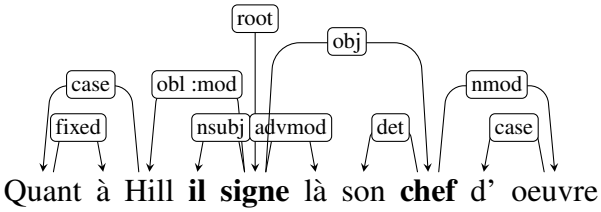

FIGURE 4.4 – Représentation du cadre `nsubj-obj` dans son contexte

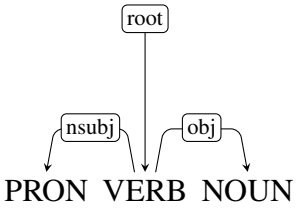

FIGURE 4.5 – Représentation du cadre `nsubj-obj` avec ses parties du discours

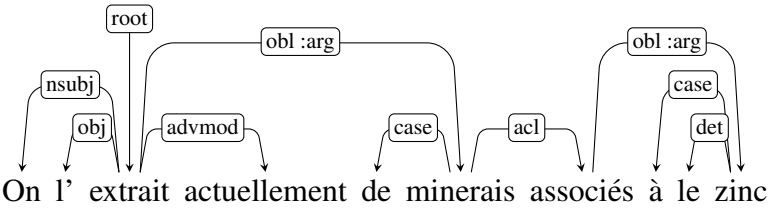

FIGURE 4.6 – Représentation du cadre `nsubj-obj-obl:arg|de` dans son contexte

4.3 Erreurs fréquentes

Notre dictionnaire est plutôt performant sur plusieurs aspects. Les cadres syntaxiques qui sont indiqués sont généralement cohérents avec les exemples donnés, et nous pouvons souvent distinguer les différents sens décernés par l'algorithme d'induction de sens. Toutefois, les erreurs qu'il contient sont nombreuses et évidentes. Voici certaines des erreurs que nous retrouvons le plus souvent.

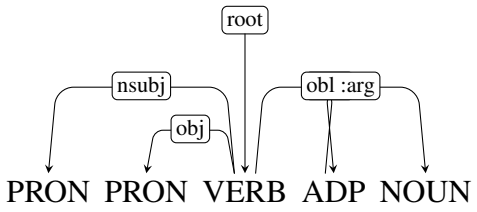

FIGURE 4.7 – Représentation du cadre `nsubj-obj-obl:arg|de` avec ses parties du discours

1. **Mauvaise identification de la partie du discours.** Le dictionnaire comporte des entrées où un sens entier correspond à un nom plutôt qu'à un verbe. Le sens 1 du verbe *extraire*, par exemple, correspond au nom *extrait* (voir exemple B.1, p. 73). Les étapes d'extraction des données et d'analyse syntaxique sont toutes deux responsables de cette erreur ; le moteur de recherche à l'aide duquel nous avons extrait nos phrases d'exemples, Sketch Engine, a à tort étiqueté ces noms comme des verbes, ce qui a introduit ces phrases dans nos données, puis l'analyseur syntaxique a analysé à son tour ces noms comme des verbes. L'étape d'induction de sens semble toutefois réussir généralement à différencier ces phrases, les rassemblant toutes dans un même sens.
2. **Mauvaise classification des phrases.** Par exemple, pour le verbe *signer* (voir B.2, p. 77), les phrases de contexte peuvent nous faire penser qu'il existe deux sens : un sens qui correspond à 'achever une œuvre', et un autre qui correspond à 'apposer sa signature à un document'. Or, plusieurs exemples contradictoires se retrouvent dans l'une et l'autre des champs "sens", ce qui indique que le regroupement s'est fait sur la base d'un critère différent, que nous ne connaissons pas.
3. **Sens difficiles à distinguer.** Il arrive que différents sens soient distingués pour un verbe, mais que la différence entre ses sens ne soit pas aisément distinguable. Le verbe *gâcher* (voir B.3, p. 78), par exemple, est divisé en deux sens. Libre au lecteur de voir s'il perçoit une différence entre les deux.
4. **Faible capacité de distinction entre ajouts et arguments.** Ce phénomène se produit surtout avec des prépositions qui précèdent typiquement des compléments de lieu. Par exemple, le verbe *insister* (voir B.4, p. 80) prend régulièrement un complément précédé de *sur*, comme dans *J'insiste bien sur ce point.* Or, ce complément est étiqueté `obl:mod`, soit comme un modificateur. Pour capturer ce type de complément, il faudrait donc garder les éléments étiquetés comme modificateurs ; or, cela inclurait du bruit dans nos données et baisserait notre score de précision.

5. **Pas de distinction entre verbes pronominaux et verbes réfléchis.** Le verbe *s'évanouir*, par exemple, n'a toujours que deux arguments : le sujet et le pronom réfléchi (voir exemple B.5, p. 83). Le pronom réfléchi ne peut être omis. Or, le sujet est alternativement étiqueté `nsubj` ou `nsubj:pass`, tandis que le pronom réflexif est étiqueté `expl:pass`, `dep:comp`, ou même `obj`. Nous retrouvons donc quatre cadres différents pour ce verbe, alors qu'en réalité il devrait n'y en avoir qu'un seul.

Chapitre 5

Conclusion

Le premier objectif de ce travail était d'extraire le sens des verbes à partir de données brutes, sans utiliser de base de sens externe, et avec une méthode qui peut s'appliquer à plusieurs langues. Pour atteindre cet objectif, nous avons extrait des plongements lexicaux contextuels à l'aide de modèles de langues de type transformeurs. Nous avons découvert que le modèle de langue qui donnait les meilleurs résultats parmi les deux modèles multilingues (*XML-RoBERTa* et *T5*) et le modèle monolingue (*CamemBERT*) testés était *CamemBERT*, ce qui laisse croire que les modèles monolingues sont plus efficaces que les modèles multilingues pour ce type de tâches. Ces plongements lexicaux ont été par la suite divisés en sens à l'aide d'un algorithme de regroupement agglomératif. En déterminant notre modèle de langue et notre algorithme de regroupement et en ajustant le paramètre du modèle à partir des données de test de FrenchSemEval, nous avons réussi à atteindre un score F_1 B^3 de 58,19 % sur un sous-ensemble des données d'entraînement, ce qui est satisfaisant selon nous pour une méthode non supervisée. Il est difficile de dire précisément comment ce score se situe par rapport à l'état de l'art ; toutefois, d'autres équipes qui se sont entraînées sur le *Wiktionnaire* et évaluées sur *FrenchSemEval*, en utilisant une méthode supervisée, ont obtenu des scores inférieurs : 43 % d'exactitude pour l'équipe de *FrenchSemEval* et 50,02 % pour l'équipe de *FlauBERT*. Ainsi, même si nous n'avons pas fait notre estimation de paramètres sur les données d'entraînement, contrairement à ces deux équipes, nos résultats démontrent le potentiel des techniques non supervisées pour l'induction du sens des verbes.

Notre deuxième objectif était d'extraire de façon automatique les cadres syntaxiques des verbes. Pour ce faire, nous avons comparé les performances de plusieurs analyseurs syntaxiques automatiques qui offrent des annotations en UD, un format d'annotation conçu pour pouvoir être utilisé sur un grand nombre de langues différentes. Nous avons déterminé que

Stanza était le meilleur choix dans notre cas, avec un score F_1 moyen de 83,29 %. Puis, à l'aide de tests manuels, nous avons déterminé un seuil de fréquence relative minimale pour distinguer les cadres syntaxiques réels du bruit. En comparant nos résultats avec les données de *Dicovalence*, une ressource créée manuellement, nous avons obtenu un score F_1 de 64,14 %, avec une précision bien supérieure au rappel (72,34 % de précision contre 57,62 % de rappel).

Enfin, nous voulions créer un dictionnaire de valence lisible à la fois par les humains et par les machines et qui distingue bien les différents sens à l'aide d'exemples. Notre dictionnaire distingue les arguments obligatoires des arguments optionnels, indique la fréquence relative de chacun des cadres, et illustre chaque sens à l'aide d'exemples.

Au terme de ce travail, nous avons donc atteint nos objectifs. Notre dictionnaire a été créé presque entièrement automatiquement (seuls les paramètres ajustables doivent être choisis par un humain, mais ceux que nous proposons peuvent certainement servir de point de départ), donne des résultats qui sont cohérents avec l'état de l'art, et est offert dans un format consultable à la fois par les humains et par les machines. La méthode que nous proposons peut être appliquée à toute autre langue qui dispose de suffisamment de données écrites. De plus, elle s'utilise avec des outils simples, ce qui la rend accessible aussi bien pour des chercheurs expérimentés que moins expérimentés.

Néanmoins, avec un score général de moins de 70 % (58,19 % pour la partie d'induction de sens, 64,14 % pour les cadres syntaxiques), le score de notre ressource est bas. Elle comporte de nombreuses erreurs, qui sont rapidement perceptibles. La division en sens laisse souvent à désirer, et les analyses syntaxiques sont souvent fautives. Utiliser la ressource telle quelle dans une application de TAL serait probablement hasardeux.

Toutefois, étant donné que nous avons fourni les fréquences relatives de chacun des cadres obtenus, il est encore possible d'améliorer notre dictionnaire. Nous avons déterminé des seuils de fréquence relative qui se voulaient conservateurs afin de favoriser la précision, mais ceux-ci ont été déterminés à partir d'un échantillon limité de 40 verbes. Il est donc toujours possible de les ajuster : on peut supposer qu'augmenter ces seuils permettrait d'augmenter la précision, tandis qu'augmenter ces seuils permettrait d'augmenter le rappel. De plus, on pourrait déterminer un nombre d'exemples minimal requis par sens afin que chaque cadre syntaxique possède au moins cinq phrases d'exemples. Cela fournirait un filtre supplémentaire et supprimerait certainement beaucoup de bruit. Enfin, chaque étape peut être améliorée au fil des mises à jour technologiques : un modèle de langue plus performant que *CamemBERT* pourrait donner de meilleurs plongements lexicaux contextuels ; d'autres algorithmes de regroupement pourraient être testés afin d'induire encore mieux les sens des verbes ; et d'autres analyseurs syntaxiques pourraient être testés.

Enfin, l'évaluation de chacune de nos tâches est une étape importante, mais délicate. Pour l'évaluation de notre induction de sens, nous avions la chance de disposer de *FrenchSemEval*, une ressource construite spécifiquement pour le français et dont la distinction de sens était plutôt conservatrice, comme la nôtre. Ce genre de ressources n'existe toutefois pas pour toutes les langues, ce qui peut rendre l'évaluation de l'induction de sens ardue dans plusieurs cas. Nous avons déterminé le paramètre le plus approprié pour le français, et nous supposons que ce paramètre donnerait des résultats adéquats pour d'autres langues, mais il faudrait vérifier cette supposition. Quant au processus d'évaluation des cadres syntaxiques, il mérite certainement d'être revu. Nous avons fait notre évaluation manuellement, mais il est possible de faire automatiquement une correspondance entre les formats d'arguments de la ressource de référence et ceux de notre ressource afin de calculer automatiquement notre score. Nous avons utilisé *Dicovalence*, qui est une ressource de qualité, qui n'est pas parfaitement adaptée à nos besoins en raison de sa couverture et de son traitement des compléments optionnels. De plus, nous avions là aussi la chance de disposer d'une ressource en français, ce qui ne sera pas le cas pour toutes les langues.

Bibliographie

AMIGÓ, E., GONZALO, J., ARTILES, J. et VERDEJO, F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12:461–486.

BAGGA, A. et BALDWIN, B. (1998). Entity-based cross-document coreferencing using the vector space model. *In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98/COLING '98, page 79–85, USA. Association for Computational Linguistics.

BEVILACQUA, M., PASINI, T., RAGANATO, A. et NAVIGLI, R. (2021). Recent Trends in Word Sense Disambiguation : A Survey. *In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4330–4338. International Joint Conferences on Artificial Intelligence Organization.

BOURIGAULT, D., FABRE, C., FRÉROT, C., JACQUES, M.-P. et OZDOWSKA, S. (2005). Syntex, analyseur syntaxique de corpus. *In Actes des 12èmes journées sur le Traitement Automatique des Langues Naturelles*, Dourdan, France.

CAMPELLO, R. J. G. B., MOULAVI, D. et SANDER, J. (2013). Density-based clustering based on hierarchical density estimates. *In Advances in Knowledge Discovery and Data Mining*, pages 160–172, Berlin, Heidelberg.

CHOMSKY, N. (1965). *Aspects of the Theory of Syntax*. The MIT Press, 50e édition.

CONNEAU, A., KHANDELWAL, K., GOYAL, N., CHAUDHARY, V., WENZEK, G., GUZMÁN, F., GRAVE, E., OTT, M., ZETTLEMOYER, L. et STOYANOV, V. (2020). Unsupervised cross-lingual representation learning at scale. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

DANLOS, L., PRADET, Q., BARQUE, L., NAKAMURA, T. et CONSTANT, M. (2016). Un verbenet du français. *Revue TAL*, 57(1):25.

DELLI BOVI, C., CAMACHO-COLLADOS, J., RAGANATO, A. et NAVIGLI, R. (2017). EuroSense : Automatic harvesting of multilingual sense annotations from parallel text. *In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, pages 594–600, Vancouver, Canada. Association for Computational Linguistics.

DUBOIS, J. et DUBOIS-CHARLIER, F. (2010). La combinatoire lexico-syntaxique dans le dictionnaire électronique des mots. les termes du domaine de la musique à titre d'illustration. *Langages*, 179-180(3–4):31–56.

DUECK, D. (2009). *Affinity Propagation : Clustering Data by Passing Messages*. Thèse de doctorat, University of Toronto.

FALK, I., FRANCOPOULO, G. et GARDENT, C. (2007). Évaluer SYNLEX. *In Actes de la 14ème conférence sur le Traitement Automatique des Langues Naturelles. Articles longs*, pages 315–324, Toulouse, France. ATALA.

GROBOL, L. et CRABBÉ, B. (2021). Analyse en dépendances du français avec des plongements contextualisés (French dependency parsing with contextualized embeddings). *In Actes de la 28e Conférence sur le Traitement Automatique des Langues Naturelles. Volume 1 : conférence principale*, pages 106–114, Lille, France. ATALA.

GROSS, M. (1975). *Méthodes en syntaxe : Régime des constructions complétives*. Hermann, Paris.

HADOUCHE, F. et LAPALME, G. (2010). Une version électronique du lvf comparée avec d'autres ressources lexicales. *Langages*, 179-180(3-4):193–220.

HOCHREITER, S. et SCHMIDHUBER, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

KIPPER, K., KORHONEN, A., RYANT, N. et PALMER, M. (2006). Extending VerbNet with novel verb classes. *In Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

KOEHN, P. (2005). Europarl : A parallel corpus for statistical machine translation. *In Proceedings of Machine Translation Summit X : Papers*, pages 79–86, Phuket, Thailand.

KORHONEN, A. (2002). Subcategorization acquisition. Rapport technique UCAM-CL-TR-530, University of Cambridge, Computer Laboratory.

KUPŚĆ, A. et ABEILLÉ, A. (2008). TreeLex : A Subcategorisation Lexicon for French Verbs. *In First International Conference on Global Interoperability for Language Resources*, Hong Kong, Hong Kong SAR China.

LE, H., VIAL, L., FREJ, J., SEGONNE, V., COAVOUX, M., LECOUTEUX, B., ALLAUZEN, A., CRABBÉ, B., BESACIER, L. et SCHWAB, D. (2019). Flaubert : Unsupervised language model pre-training for french. *arXiv preprint arXiv :1912.05372*.

LENCI, A., LAPESA, G. et BONANSINGA, G. (2012). LexIt : A computational resource on Italian argument structure. *In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3712–3718, Istanbul, Turkey. European Language Resources Association (ELRA).

LIN, D. (1998). Automatic retrieval and clustering of similar words. *In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*, ACL '98/COLING '98, pages 768–774, USA. Association for Computational Linguistics.

LIU, Y., OTT, M., GOYAL, N., DU, J., JOSHI, M., CHEN, D., LEVY, O., LEWIS, M., ZETTLEMOYER, L. et STOYANOV, V. (2019). Roberta : A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

MARTELLI, F., KALACH, N., TOLA, G. et NAVIGLI, R. (2021). SemEval-2021 task 2 : Multilingual and cross-lingual word-in-context disambiguation (MCL-WiC). *In Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 24–36, Online. Association for Computational Linguistics.

MARTIN, L., MULLER, B., ORTIZ SUÁREZ, P. J., DUPONT, Y., ROMARY, L., de la CLERGERIE, É., SEDDAH, D. et SAGOT, B. (2020). CamemBERT : a tasty French language model. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

MESSIANT, C., POIBEAU, T. et KORHONEN, A. (2008). LexSchem : a large subcategorization lexicon for French verbs. *In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

MILLER, G. A. (1992). WordNet : A lexical database for English. *In Speech and Natural Language*, Harriman, New York.

MILLER, G. A., LEACOCK, C., TENGI, R. et BUNKER, R. T. (1993). A semantic concordance. *In Human Language Technology*, Plainsboro, New Jersey.

NAVIGLI, R. et PONZETTO, S. P. (2012). BabelNet : The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

NIVRE, J., de MARNEFFE, M.-C., GINTER, F., GOLDBERG, Y., HAJIČ, J., MANNING, C. D., MCDONALD, R., PETROV, S., PYYSALO, S., SILVEIRA, N., TSARFATY, R. et ZEMAN, D. (2016). Universal Dependencies v1 : A multilingual treebank collection. *In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

OSBORNE, T. et GERDES, K. (2019). The status of function words in dependency grammar : A critique of universal dependencies (ud). *Glossa : a journal of general linguistics (2016-2021)*.

PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M. et DUCHESNAY, E. (2011). Scikit-learn : Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

QI, P., ZHANG, Y., ZHANG, Y., BOLTON, J. et MANNING, C. D. (2020). Stanza : A Python natural language processing toolkit for many human languages. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics : System Demonstrations*.

RAFFEL, C., SHAZEER, N., ROBERTS, A., LEE, K., NARANG, S., MATENA, M., ZHOU, Y., LI, W. et LIU, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer.

RAGANATO, A., CAMACHO-COLLADOS, J. et NAVIGLI, R. (2017). Word sense disambiguation : A unified evaluation framework and empirical comparison. *In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 1, Long Papers*, pages 99–110, Valencia, Spain. Association for Computational Linguistics.

RAMBELLI, G., LEBANI, G., PRÉVOT, L. et LENCI, A. (2016). LexFr : Adapting the LexIt framework to build a corpus-based French subcategorization lexicon. *In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 930–937, Portorož, Slovenia. European Language Resources Association (ELRA).

RAMBELLI, G., LENCI, A. et POIBEAU, T. (2017). UDLex : Towards cross-language subcategorization lexicons. *In Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*, pages 207–2017, Pisa,Italy. Linköping University Electronic Press.

SAGOT, B. (2010). The lefff, a freely available and large-coverage morphological and syntactic lexicon for French. *In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

SCHMID, H. (1995). Improvements in part-of-speech tagging with an application to german. *In Proceedings of the ACL SIGDAT-Workshop*.

SCHÜTZE, H. (1998). Automatic word sense discrimination. *Comput. Linguist.*, 24(1): 97–123.

SEGONNE, V., CANDITO, M. et CRABBÉ, B. (2019). Using Wiktionary as a resource for WSD : the case of French verbs. *In Proceedings of the 13th International Conference on Computational Semantics – Long Papers*, pages 259–270, Gothenburg, Sweden. Association for Computational Linguistics.

SIMON, E., SERÉNY, A. et BABARCZY, A. (2010). Automatic acquisition of hungarian subcategorization frames. *Methods for the automatic acquisition of Language Resources and their evaluation methods*, page 13.

SZEKELY, G. et RIZZO, M. (2005). Hierarchical clustering via joint between-within distances : Extending ward's minimum variance method. *Journal of Classification*, 22:151–183.

TESNIÈRE, L. (1959). *Élements de syntaxe structurale*. C. Klincksieck, Paris.

van den EYNDE, K. et BLANCHE-BENVENISTE, C. (1978). Syntaxe et mécanismes descriptifs : Présentation de l'approche pronominale. *In Cahiers de lexicologie*, volume 32, pages 3–27.

van den EYNDE, K. et MERTENS, P. (2003). La valence : l'approche pronominale et son application au lexique verbal. *Journal of French Language Studies*, 13(1):63–104.

VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, L. et POLOSUKHIN, I. (2017). Attention is all you need. *In Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

VÉRONIS, J. (2004). Hyperlex : Lexical cartography for information retrieval. *Computer Speech & Language*, 18:223–252.

ZEMAN, D., NIVRE, J. *et al.* (2021). Universal dependencies 2.9. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

ZIEMSKI, M., JUNCZYS-DOWMUNT, M. et POULIQUEN, B. (2016). The United Nations parallel corpus v1.0. *In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).

Annexe A

Extrait du *Dicovalence*

Listing A.1 – Entrées pour le verbe *rapprocher* dans *Dicovalence*

```
VAL$  rapprocher: P0 P1 P3
VTYPE$  predicator simple
VERB$ RAPPROCHER/rapprocher
NUM$  66890
EG$ il nous rapproche de cet objectif
TR_DU$  bijeenbrengen, bijeenplaatsen, (nader) tot elk brengen,
   verzoenen, vergelijken, naast elk stellen, in verband brengen
TR_EN$
FRAME$  subj:pron|n:[hum], obj:pron|n:[hum,nhum,?abs],
   objde:pron|n:[hum,nhum,?abs]
P0$ que, qui, je, nous, elle, il, ils, on, ça, celui-ci, ceux-ci
P1$ que, qui, te, vous, la, le, les, se réfl., en Q, ça, ceci,
   celui-ci, ceux-ci
P3$ quoi, qui, en, lui_ton, eux, ça, ceci, celui-ci, ceux-ci
LCCOMP$ je rapproche celui-ci de celui-là, je les rapproche
LC$ 66890-66920 je rapproche celui-ci de celui-là, celui-ci se
   rapproche de celui-là
AUX$  avoir


VAL$  rapprocher: P0 P1
VTYPE$  predicator simple
VERB$ RAPPROCHER/rapprocher
```

```
NUM$  66891
EG$ les journalistes ont rapproché les deux faits
TR_DU$  bijeenbrengen, bijeenplaatsen, (nader) tot elk brengen,
   verzoenen, vergelijken, naast elk stellen, in verband brengen
TR_EN$  link, connect, interrelate
FRAME$  subj:pron|n:[hum], obj:pron|n:[hum,+complex]
P0$ que, qui, je, nous, elle, il, ils, on, ça, celui-ci, ceux-ci
P1$ que, qui, vous, les, en Q, ça, ceux-ci
LCCOMP$ je rapproche celui-ci de celui-là, je les rapproche
AUX$  avoir


VAL$  rapprocher: P0 P1 (PP<de>)
VTYPE$  predicator simple
VERB$ RAPPROCHER/rapprocher
NUM$  66900
EG$ il rapproche son fauteuil de la télé
TR_DU$  dichterbij brengen
TR_EN$  bring nearer (to)
FRAME$  subj:pron|n:[hum], obj:pron|n:[nhum,?abs],
   ?objp<de>:pron|n:[nhum,?abs]
P0$ que, qui, je, nous, elle, il, ils, on, ça, celui-ci, ceux-ci
P1$ que, (qui), (te), (vous), la, le, les, en Q, ça, ceci,
   celui-ci, ceux-ci
PP_PR$  de
PP$ 0, quoi, ça, ceci, celui-ci, ceux-ci
RP$ passif être, se passif
AUX$  avoir


VAL$  se rapprocher: P0 (P3)
VTYPE$  predicator simple pseudo_se
VERB$ RAPPROCHER/rapprocher
NUM$  66910
EG$ elle se rapproche de son but
TR_DU$  naderbij komen, naderen
TR_EN$  near
```

```
FRAME$  pseudo_se, subj:pron|n:[hum,nhum,?abs],
   ?objde:pron|n:[hum,nhum,?abs]
P0$ que, qui, je, nous, elle, il, ils, on, ça, ceci, celui-ci,
   ceux-ci
P3$ 0, quoi, qui, en, lui_ton, eux, ça, ceci, celui-ci, ceux-ci
AUX$  être


VAL$  se rapprocher: P0 P3
VTYPE$  predicator simple pseudo_se
VERB$ RAPPROCHER/rapprocher
NUM$  66920
EG$ il s'est rapproché des descriptivistes
TR_DU$  naderbij komen, zich verzoenen (met)
TR_EN$  come nearer (to), reconcile/conciliate (with)
FRAME$  pseudo_se, subj:pron|n:[hum], objde:pron|n:[hum,abs]
P0$ que, qui, je, nous, elle, il, ils, on, ça, celui-ci, ceux-ci
P3$ quoi, qui, en, lui_ton, eux, ça, celui-ci, ceux-ci
LC$ 66890-66920 je rapproche celui-ci de celui-là, celui-ci se
   rapproche de celui-là
LCCOMP$ il se rapproche de lui, ils se rapprochent
AUX$  être


VAL$  se rapprocher: P0
VTYPE$  predicator simple pseudo_se
VERB$ RAPPROCHER/rapprocher
NUM$  66925
EG$ après leur dispute ils se sont rapprochés à nouveau
TR_DU$  zich verzoenen (met), gelijkenis vertonen, vergeleken
   kunnen worden
TR_EN$  reconcile, reconciliate; be comparable (with)
FRAME$  pseudo_se, subj:pron|n:[hum,+complex]
P0$ que, qui, nous, ils, on, ça, ceux-ci
LCCOMP$ il se rapproche d'eux, ils se rapprochent
AUX$  être
```

Annexe B

Extraits de notre dictionnaire

Listing B.1 – Entrée du verbe *extraire*


```
"extraire": {
 "sens": [
  [{"nom": "nsubj-obl:arg|de",
    "freq": 0.39,
    "exemples": [
     "Anton Rubinstein Extrait de Le démon",
     "Jean Tardieu Extrait de Un Mot pour un autre",
     "Michel REVERCHON-BILLOT EVS Extrait de la présentation d
        ouverture du printemps de l EMI (30 mars 2016) 1 LES
        TEXTES DE REFERENCE.",
     "Hélène CIXOUS Extrait du programme du spectacleTexte
        intégral en introduction de L'Indiade ou l'Inde de
        leurs rêves, Théâtre du Soleil, 1987, pp. 11-17et dans
        Acteurs, nÂ°53, octobre 1987, pp. 8-9"],
    "sous-cadres": []},
   {"nom": "nsubj-obj",
    "freq": 0.15,
    "exemples": [
     "MAE, Paris, CC, vol. 36, doc. 92, Extrait des minutes de
        la chancellerie du consulat de la République française à
         Tunis, décembre 1796.",
     "Extrait du multivision 'Un Parc comme un écrin' -
        Réalisation Parc national des Ecrins"],
```

```
   "sous-cadres": []},
  {"nom": "obj",
   "freq": 0.23,
   "exemples": [
    "[ Extrait des dernières paroles de Périclite, général
       grec, à la veille de la bataille de Chéronée]",
    "1375. - Extrait des lettres patentes pour justifier du
       pouvoir que les consuls ont de contraindre les
       ecclésiastiques au payement des tailles et aux frais de
       la perte.",
    "Mise au point - le traitement hormonal de la ménopause,
       point d'étape, juin 2006, Afssaps.> Brière, J., 10
       mythes concernant la ménopause, Ressources santé,
       2002.> André, J., Le traitement des bouffées de
       chaleur, Extrait des Mises à jour en Gynécologie
       Médicale, 2002."],
   "sous-cadres": []}
 ],
 [{"nom": "nsubj-obj",
   "freq": 0.7,
   "exemples": [
    "A partir de l'échantillon biologique à tester, on extrait
       les ARN messagers totaux que l'on transforme en ADN
       complémentaires marqués.",
    "De la partie grasse des graines, on extrait des
       lécithines (émulsifiants(E322) très employées pour
       toutes sortes de produits de l'industrie
       agro-alimentaire et des acides gras essentiels
       incorporés, entre autres, dans la fabrication des laits
       pour nourrissons.",
    "Enfin, on extrait l'url de chaque ligne restante.",
    "On extrait la racine carrée de 654 324 qui est
       approximativement 808,9.",
    "Si le numéro obtenu n'est pas un multiple de 3, on
       extrait une boule dans l'urne 2 qui contient 3 boules
       noires et 2 boules blanches."],
   "sous-cadres": [
```

```
    {"nom": "nsubj-obj-obl:arg|de",
     "freq": 0.07,
     "exemples": [
      "D'un bond, il s' extrait du lit et regarde par la
         fenêtre.",
      "On l' extrait maintenant des plaines de Bagneux et
         d'Arcueil; on en tire aussi de Saint-Denis.",
      "On l' extrait actuellement de minerais associés au zinc
         (la blende), à l'argent et (le plus abondamment) au
         cuivre."
     ]},
    {"nom": "nsubj-obj-obl:arg",
     "freq": 0.08,
     "exemples": [
      "On en extrait de l'huile soit à usage alimentaire, é
         nergétique voire même industriel.",
      "En chauffant cette dernière, on en extrait le gaz
         ammoniacB.",
      "On en extrait l'huile de jojoba, une sorte de cire
         liquide comparable au sébum et qui ne rancit pas.",
      "On en extrait plusieurs quarante-cinq tours dont deux,
         Pourquoi faut-il et Un jour comme les autres ainsi
         que Cendrillon et Tout va trop vite qui figureront
         dans la prestigieuse Série Gémini de CBS France [4]."
     ]}
   ]
  }
],
[{"nom": "nsubj-obj",
  "freq": 0.62,
  "exemples": [
```

```
 "Nous en extrayons les renseignements suivants : ils
    embrassent la période 1881-1893, car M.
    Dujardin-Beaumetz avait joint à son rapport de 1891
    \"\" une vue d'ensemble sur les faits d'hydrophobie
    constatés dans la ville de Paris et dans le département
    de la Seine, depuis le premier rapport qu'il a adressé
    au Préfet de Police, c'est-à-dire depuis 1881.",
 "La qualité et limportance de ces travaux extraient de
    nombreux problèmes équestres de tout occultisme
    empirique et explosent les idées fausses ou reçues.",
 "Répondant à ses questions, il extraie son propre cœur et
    montre à quel point les ténèbres l'ont obscurci.",
 "Cependant si, des informations de la Langue Sacrée, nous
    extrayons toujours l'Esprit de la Prophétie', prophétie
    qui se confirme sans cesse au cours des temps en ceux
    qui cherchent la Gnose, alors tout ce qui est vague et
    incomplet tombe immédiatement et nous possédons un
    trésor, une abondance de données, dindications et de
    directives.",
 "Dans le processus de la recontextualisation, les
    participants extraient une partie du texte, des signes
    ou du sens du contexte original (décontextualisation)
    pour les introduire dans un autre contexte."],
"sous-cadres": [
 {"nom": "nsubj-obj-obl:arg",
  "freq": 0.04,
  "exemples": [
   "Nous en extrayons sous forme d'analyse les passages
      suivants :",
   "Joly, voyant un chat rôder sur une gouttière, en
      extrayait la phi losophie.",
   "Nous en extrayons les passages les plus significatifs
      suivants :",
   "Ils en extraient le miel à la belle saison.",
   "J'en extrais cette lettre, datée du 6 septembre 1973."
  ]},
 {"nom": "nsubj-obj-obl:arg|de",
```

```
      "freq": 0.07,
      "exemples": [
       "Au Moyen Âge, pour la préparation du cuir (tannerie),
          on l' extrayait d'écorce de chêne qu'on broyait dans
          des moulins à tan.",
       "On extraira de ce mémoire pour la Connaissance des tems
          la table de la hauteur moyenne du baromètre dans 124
          villes de la France.",
       "Oubliée par la médecine moderne, on extrayait de ses
          racines une substance permettant d'augmenter
          l'activité biliaire et comme plante purgative des
          intestins."
      ]}
    ]
   }
  ]
 ]
},
```

Listing B.2 – Entrée du verbe *signer*

```
"signer": {
 "sens": [
  [{"nom": "nsubj-obj",
    "freq": 0.73,
    "exemples": [
     "Afin de remplacer le T-33, le Portugal signa au milieu
        des années 1980 une lettre d'intention avec la Skyfox
        Corporation pour l'achat de vingt kits de conversion.",
     "Le comte de Toulouse Raimon VII et le roi de France Louis
        IX signent un traité de paix à Lorris dans le Loiret.",
     "Donc, quand ils peuvent prendre un dossier, ils le
        signent pour augmenter leur CA.",
     "En 2002, ils signent un contrat avec le label allemandÂ :
        Euphonic Records (le label de Ronski Speed).",
     "Il signait 'capitaine' justement pour s'imposer devant
        les autorités."],
    "sous-cadres": []}
```

```
    ],
    [{"nom": "nsubj-obj",
      "freq": 0.84,
      "exemples": [
       "Le 23 septembre 2016, il signe un contrat non garanti
          avec le Thunder d'Oklahoma City pour participer au camp
          d'entraînement et tenter de faire partie des quinze
          joueurs retenus au début de la saison NBA
          2016-2017[1].",
       "Quant à Hill il signe là son chef d'œuvre, un film é
          légiaque d'une rare mélancolie.",
       "En mars 2014, il signe un contrat espoir avec la province
          du Munster à partir de la saison 2014-2015[6].",
       "Avec 'Le survivant' il signe un roman très différent,
          mais considéré par beaucoup comme un chef-d'oeuvre.",
       "Après un essai non concluant au Stade brestois[13] et
          dans les club belges du RAEC Mons en août[14], puis du
          VfR Aalen en septembre[15], il signe lors du mois de
          décembre 2008 à Leeds United un contrat d'un
          mois[16],[17]."],
      "sous-cadres": []}
    ]
   ]
  },
```

Listing B.3 – Entrée du verbe *gâcher*

```
  "gâcher": {
   "sens": [
    [{"nom": "nsubj-obj",
      "freq": 0.55,
      "exemples": [
       "Par ailleurs, elle ne gâchera pas votre décoration, au
          contraire, elle donnera une touche plus design à votre
          cuisine.",
       "Vous en dévoiler plus vous gâcherait le plaisir de
          découvrir cet album.",
```

```
    "\"Source de l'article: Acouphènes : Comment se soigner ?
       - vidéo dailymotion. \"\"Existe-t-il un traitement
       contre les acouphènes ?\"\" Les acouphènes\" gâchent la
       vie de centaines de milliers de nos compatriotes.",
    "Pourquoi gâchait-on notre jeune enthousiasme à coup de
       corvées de sanctions ?",
    "Rougeurs, yeux embrumés, éternuements... Les allergies
       gâchent le quotidien de nombreuses personnes à travers
       le monde."],
   "sous-cadres": []},
  {"nom": "obj",
   "freq": 0.1,
   "exemples": [
    "â†' Â«Â Azarenka gâche la fêteÂ Â», sur
       lequipe.fr(consulté le 17 février 2013)",
    "À trop jouer le secret Apple gâchera \" l'expérience
       utilisateur des clients les premiers mois si ils
       passent vraiment l'écran en 4\"\"\"",
    "Alors ne gâchons pas nos existences et tentons de vivre à
       la hauteur de ces ancêtres qui ont su produire des
       chefs d'œuvre dans un monde qui n'était qu'hostilité.",
    "Ne la gâchons pas, ne nous dérobons pas.",
    "Ne la gâchez pas en vous amusant à critiquer la
       concurrence uniquement pour vous faire bien voir par
       votre potentiel recruteur."],
   "sous-cadres": []}
 ],
 [{"nom": "nsubj-obj",
   "freq": 0.69,
   "exemples": [
    "Mais, à la suite d'une querelle, il gâche \" ses vœux
       pour des vétilles, alors que son désir le plus cher
       aurait été d'avoir un enfant, même \"\"haut comme un
       pouce\"\".\"",
    "L'ignorance gâche l'oeuvre d'une certaine façon, mais
       ceci n'est pas une raison de ne pas l'apprécier, bien
       heureusement, donc bien joué si ça marche !",
```

```
      "Le Canada gâche \" en partie le potentiel de ces
         immigrés. \"\"\"",
      "Les gens prêts a s'investir ne gâche pas toutes leurs
         relation mais en coréens c'est durs d'en avoir.",
      "Sinon ça gâche ses effets de manches retroussées."],
     "sous-cadres": []}
   ]
  ]
 }
```


Listing B.4 – Entrée du verbe *insister*


```
 "insister": {
  "sens": [
   [{"nom": "nsubj",
     "freq": 0.81,
     "exemples": [
      "Kanner insiste sur la spécificité de ce symptôme et a le
         souci d'en faire un syndrome clinique à part entière
         spécifiant que son mode d'apparition et son évolution
         sont radicalement distincts de ceux de la
         schizophrénie.",
      "J' insiste sur le fait que je préfère répondre aux
         personnes qui ont un profil dont les informations
         principales sont renseignées.",
      "Après avoir savouré et admiré, le client insiste pour que
         le barman appelle ce cocktail 'Alexander', en hommage à
         son fils qui était né la veille : 'Alexander Lowen'.",
      "â€"et dix années plus tard, il insiste pour faire valoir
         cette péripétie où il a été ridiculisé et renvoyé vers
         le néant d'où il n'aurait jamais dû s'échapper.",
      "J' insiste aussi sur le fait que la suppression des
         gardes en nuit profonde éloignera considérablement le
         malade potentiel du médecin, et cela pour une raison
         bien simple : dans les territoires ruraux et de
         montagne, il faut raisonner non pas en kilomètres, mais
         en temps de parcours."],
     "sous-cadres": []}
```

```
],
[{"nom": "nsubj",
  "freq": 0.82,
  "exemples": [
   "â†' Le gouvernement britannique insista cependant sur un
      changement dans le texte afin d'augmenter les
      possibilités d'appel au Conseil privé à Londres.",
   "Nous insistons sur le CETA, car ce traité avec le Canada,
      outre ses similitudes avec le TAFTA, constitue un
      véritable marchepied pour le TAFTA, de par
      l'imbrication des économies canadienne et é
      tats-unienne, et permettra ainsi aux entreprises
      américaines de se saisir des outils mis à disposition
      du CETA (arbitrage d'investissement, coopération
      réglementaire) pour s'attaquer aux lois et normes de la
      France et de l'UE.",
   "Dans leur vision prospective et pour répondre à
      l'évolution des menaces, les armées occidentales
      insistent sur l'importance de la mobilité et de
      l'agilité des forces et sur l'impérieuse nécessité de
      leur dispersion.",
   "Le 3 août il insista à nouveau calmement sur ce point, ce
      qui provoqua une dispute enflammée avec le chef
      d'état-major polonais, le général Rozwadowski.",
   "Les habitants de Koufa insistent pour que Husayn vienne
      les rejoindre."],
  "sous-cadres": []
 }
],
[{"nom": "nsubj",
  "freq": 0.39,
  "exemples": [
   "'Mais sans clarté', insiste le militant. 'NKM est très
      agressive.'",
   "'Too' insistent, aussi, avait une guitare un peu
      obsédante.",
```

```
   "’Je ne mets pas en cause votre sincérité, ne mettez pas
      en cause ma moralité.’ Il insiste : ’Vous sortez de vos
      gonds avec beaucoup de facilité.’",
   "’Nous demandons toujours une enquête indépendante qui
      permettrait de démasquer les commanditaires de son
      assassinat, mais je vois des signaux positifs dans le
      fait que les tireurs ont été arrêtés et que les
      investisseurs néerlandais et finlandais se soient
      retirés du projet.’ L’ONG insiste : si elles peuvent
      nous paraître lointaines, ces violences nous concernent
      tous et il est ’vital’ de protéger les défenseurs de
      l’environnement, lanceurs d’alerte terriblement
      exposés.",
   "’Et d’autre part , cela suppose un immense effort
      d’information de l’opinion publique , d’éducation de la
      jeunesse. ’La bêtise insiste toujours’, a écrit Albert
      Camus."],
  "sous-cadres": []
 },
 {"nom": "ccomp-nsubj",
  "freq": 0.1,
  "exemples": [
   "’À l’image des chargés de plaidoyer du CCFD qui ont
      multiplié les rendez-vous avec des responsables
      politiques, les étudiants, – un maillon de la chaîne
      parmi d’autres’, insiste Lara – vont mener une action
      de plaidoyer au sein de l’université. ’Nous ciblons les
      étudiants pour les amener à une prise de conscience.’",
   "Elle insiste ’Ce fut un grand choc pour moi quand je l’ai
      vu la façon dont les choses tournaient...une série de
      victoires choquantes dans des Etats clés comme la
      Floride et l’Ohio...’ Choquantes... ?",
   "La chanson Just believe insiste : ’ pour avancer il faut
      croire en ses rêves, en soi-même et en ce que l’on fait
      ’."],
  "sous-cadres": []
 }
```

```
    ]
   ]
  },
```

Listing B.5 – Entrée du verbe *(s')évanouir*

```
  "évanouir": {
  "sens": [
   [{"nom": "expl:pass-nsubj:pass",
     "freq": 0.43,
     "exemples": [
      "Dès la première parcelle de celluloïd, les doutes s' é
         vanouissent : nul besoin de discourir, cette
         excentricité-là est un enfant des studios de Morimoto.",
      "Toutes deux s' évanouissent au moment où un délégué des
         inspecteurs du travail arrive pour remettre un prix à
         Chipoutchine, mais, voyant la situation dans le bureau,
         il repart avec son prix.",
      "L'amertume ne s' évanouirait pas de sitôt, mais la
         procédure de rapatriement suivit son cours.",
      "Ainsi, Dieu est en même temps la foi et le garde-fou de
         la pensée humaine ; quand il est chassé de
         l'intelligence d'une nation, la place qu'il occupait ne
         tarde pas d'être envahie par une sorte de possession
         infernale ; à sa suite, l'art, la poésie, l'amour, le
         courage, le génie s' évanouissent , les croyants sont
         remplacés par des monstres, et l'apostasie est punie
         par l'abrutissement.",
      "Ses illusions s' évanouissent quand la jeune fille se
         fiance avec un commerçant de Leipzig."],
     "sous-cadres": []},
    {"nom": "expl:pass-nsubj",
     "freq": 0.12,
     "exemples": [
      "Quatre de ces femmes s' évanouirent ... tombèrent, et ne
         furent relevées qu'à coups de fouet.",
      "Soudain, bruits et lueur s' évanouirent .",
```

```
    "Tout s’en allait soudain de lui, comme une obscure fumée;
       la bonne grâce qu’il trouvait en Giulia, s’ é
       vanouissait .",
    "Contre terre beaucoup s’ évanouissent (La Chanson de
       Roland).",
    "Les gens, très fatigués, s’ évanouissaient les uns après
       les autres et plus personne n’avait envie de
       travailler."],
   "sous-cadres": []}
],
[{"nom": "expl:pass-nsubj:pass",
   "freq": 0.18,
   "exemples": [
    "Enfin la frustration de Thierry s’ évanouit .",
    "Son corps s’ évanouit en un clin d’œil pour se
       matérialiser devant chez elle.",
    "Enfin, le fantôme s’ évanouit .",
    "Parvenu à Castets-en-Dorthe, la marée montante s’ é
       vanouit .",
    "Une civilisation de type nettement agraire et modérément
       stra tifiée s’ évanouit et donne naissance à une a utre
       civilisation plus ou moins marquée par le début de
       l’urbanisme qui modifie radicalement les comportements
       sociaux (2)."],
   "sous-cadres": []},
  {"nom": "dep:comp-nsubj",
   "freq": 0.25,
   "exemples": [
    "Ainsi s’ évanouit la dernière espérance du repos de Dieu
       sur la terre.",
```

```
   "– L’ingrate ! ce matin encore elle partageait au château
      les biscuits de mon déjeuner. – Depuis dix ans,
      monseigneur, elle vient dans mon cachot partager mon
      pain noir. – Jour-de-Dieu ! murmura le jeune prince...
      Mais sa colère enfantine s’ évanouit devant un sourire
      malicieux de Nemours. ’Je crois, monseigneur, dit le
      jeune duc, que vous me feriez volontiers l’honneur de
      rompre une lance avec moi pour les beaux yeux d’une
      souris.’",
   "Elle s’ évanouit .",
   "Jan, qui souffre de troubles cardiaques, s’ évanouit et
      sa femme le réconforte, après lui avoir donné un
      médicament.",
   "puis il s’ évanouit face contre terre, jérémie alla
      aussitôt le voir avec anthéa pendant que frranz essaya
      d’annoncer la nouvelle aux autres."],
  "sous-cadres": []},
 {"nom": "expl:pass-nsubj",
  "freq": 0.15,
  "exemples": [
   "Au Brésil, la tolérance s’ évanouit avec le temps, de
      même que l’évangélisme progresse.",
   "Sa mère s’ évanouit .",
   "La chance qu’elle puisse passer inaperçue dans les
      ténèbres s’ évanouit .",
   "Ten Shin Han, à bout de force, s’ évanouit mais ne meurt
      pas, ce qui prouve l’évolution de sa puissanceÂ : par
      le passé, un seul Kikoho mettait sa vie en danger,
      tandis que durant la saga de Cell, il est capable d’en
      enchaîner plusieurs sans mourir[dbz_ep 6].",
   "Le renne s’ évanouit et lorsqu’il se réveilla, il était
      dans une maison, soigné et en pleine santé."],
  "sous-cadres": []},
 {"nom": "nsubj-obj",
  "freq": 0.13,
  "exemples": [
```

```
      "Cette fois, je l’avoue, la chose dépasse les rêves les
         plus fantaisistes ! – Voici la nourriture que prend,
         une ou deux fois la semaine, Hadaly, répondit Edison.
         [...] lorsqu’elle ne trouve pas ces aliments sous sa
         main au moment où elle les désire, elle s’ évanouit ou,
         pour mieux dire, elle meurt. – Elle meurt ?... murmura
         le jeune lord en souriant. – Oui, pour donner à son élu
         le plaisir vraiment divin de la ressusciter.",
      "Parce que presque tout – les attentes, la fierté, la peur
         de l’embarras ou de l’échec –, tout cela s’ évanouit
         face à la mort.",
      "Lors d’une messe, Paola s’ évanouit et John lui porte
         secours : ce contact, à la fois br (...)",
      "Découvrant que sa fille est non seulement mariée, mais
         que son époux est un vieil excentrique bien plus âgé
         qu’elle, Marietta s’ évanouit ."],
     "sous-cadres": []}
  ]
 ]
},
```

Annexe C

Verbes présents dans notre dictionnaire

- abandonner
- abattre
- abolir
- abonder
- abonner
- aborder
- aboutir
- abriter
- abroger
- abréger
- absorber
- abstenir
- abstraire
- abuser
- abîmer
- accabler
- accentuer
- accepter
- accidenter
- accoler
- accommoder
- accompagner
- accomplir
- accorder
- accoucher
- accrocher
- accroître
- accréditer
- accueillir
- accumuler
- accuser
- accéder
- accélérer
- acharner
- acheminer
- acheter
- achever
- acquitter
- acquérir
- acter
- actionner
- activer
- actualiser
- adapter
- additionner
- adhérer
- adjoindre
- admettre
- administrer
- admirer
- adonner
- adopter
- adorer
- adosser
- adoucir
- adresser
- advenir
- affaiblir
- affamer
- affecter
- affectionner
- afficher
- affilier
- affiner
- affirmer
- affranchir
- affronter
- agacer
- agencer

- aggraver
- agir
- agiter
- agrandir
- agresser
- agréer
- agréger
- agrémenter
- aider
- aiguiser
- ailler
- aimer
- ajouter
- ajuster
- alcooliser
- alerter
- aligner
- alimenter
- allaiter
- aller
- allier
- allonger
- allouer
- allumer
- alléger
- alourdir
- alterner
- altérer
- amener
- aminer
- amorcer
- amortir
- amplifier
- amputer
- amuser
- améliorer
- aménager
- analyser
- ancrer
- animer
- annexer
- annoncer
- annuler
- anticiper
- anéantir
- apaiser
- apercevoir
- aplatir
- apparaître
- apparenter
- appartenir
- appeler
- applaudir
- appliquer
- apporter
- apposer
- apprendre
- apprivoiser
- approcher
- approfondir
- approprier
- approuver
- approvisionner
- apprécier
- appréhender
- apprêter
- appuyer
- arbitrer
- arborer
- archiver
- argenter
- argumenter
- armer
- arpenter
- arracher
- arranger
- arriver
- arrondir
- arroser
- arrêter
- articuler
- aspirer
- assaisonner
- assassiner
- assembler
- asseoir
- assigner
- assimiler
- assister
- assiéger
- associer
- assortir
- assouplir
- assujettir
- assumer
- assurer
- attacher
- attaquer
- attarder
- atteindre
- atteler
- attendre
- atterrir
- attester
- attirer

- attiser
- attraper
- attribuer
- atténuer
- auditionner
- augmenter
- automatiser
- autoriser
- avaler
- avancer
- aventurer
- avertir
- aviser
- avoir
- avoisiner
- avorter
- avouer
- avérer
- axer
- aérer
- baigner
- baiser
- baisser
- balader
- balancer
- balayer
- baliser
- bander
- bannir
- banquer
- baptiser
- barrer
- basculer
- baser
- battre
- bercer
- blanchir
- blesser
- blinder
- bloquer
- boire
- boiser
- boiter
- bombarder
- bomber
- bondir
- border
- borner
- bosser
- boucher
- boucler
- bouffer
- bouger
- bouillir
- bouleverser
- bourrer
- bousculer
- brancher
- brandir
- branler
- brasser
- braver
- breveter
- bricoler
- briller
- briser
- brocher
- broder
- brosser
- brouiller
- broyer
- brûler
- buter
- bâtir
- bénir
- bénéficier
- cacher
- cadrer
- calculer
- caler
- calmer
- cambrer
- camer
- camper
- canaliser
- cantonner
- capitaliser
- capter
- capturer
- caractériser
- carburer
- caresser
- carrer
- cartonner
- caser
- casser
- castrer
- causer
- censurer
- centraliser
- centrer
- cerner
- certifier
- cesser
- changer

- chanter
- charger
- charmer
- chasser
- chauffer
- chausser
- cheminer
- chercher
- chevaucher
- chier
- chiffrer
- chiner
- choir
- choisir
- choquer
- chromer
- chuter
- chérir
- cibler
- circuler
- ciseler
- citer
- clamer
- claquer
- clarifier
- classer
- clicher
- climatiser
- cliquer
- clore
- clôturer
- cocher
- coder
- codifier
- coexister
- cohabiter
- coiffer
- coincer
- collaborer
- collecter
- collectionner
- coller
- coloniser
- colorer
- combattre
- combiner
- combler
- commander
- commencer
- commenter
- commercialiser
- commettre
- communiquer
- commémorer
- comparaître
- comparer
- compenser
- compiler
- compliquer
- compléter
- comporter
- composer
- comprendre
- compresser
- comprimer
- compromettre
- comptabiliser
- compter
- concentrer
- concerner
- concerter
- concevoir
- concilier
- conclure
- concocter
- concourir
- concrétiser
- concurrencer
- concéder
- condamner
- condenser
- conditionner
- conduire
- confectionner
- confesser
- confier
- configurer
- confiner
- confire
- confirmer
- confisquer
- confondre
- conformer
- conforter
- confronter
- conférer
- congeler
- conjoindre
- conjuguer
- connaître
- connecter
- conquérir
- consacrer
- conseiller
- consentir

- conserver
- considérer
- consigner
- consister
- consoler
- consolider
- consommer
- conspirer
- constater
- constituer
- construire
- consulter
- consumer
- contacter
- contaminer
- contempler
- contenir
- contenter
- conter
- contester
- continuer
- contourner
- contracter
- contraindre
- contrarier
- contraster
- contredire
- contrer
- contribuer
- controverser
- contrôler
- convaincre
- convenir
- converger
- convertir
- convier
- convoiter
- convoquer
- coopérer
- coordonner
- copier
- correspondre
- corriger
- corrompre
- corser
- coter
- cotiser
- coucher
- coudre
- couler
- couper
- coupler
- courber
- courir
- couronner
- couvrir
- coïncider
- coûter
- cracher
- craindre
- craquer
- creuser
- crever
- crier
- cristalliser
- critiquer
- croire
- croiser
- croquer
- croître
- créditer
- créer
- cuber
- cueillir
- cuire
- cuisiner
- culminer
- cultiver
- cumuler
- curer
- céder
- célébrer
- côtoyer
- daller
- danser
- dater
- demander
- demeurer
- descendre
- desservir
- dessiner
- destiner
- devancer
- devenir
- deviner
- devoir
- diagnostiquer
- dialoguer
- dicter
- diffuser
- différencier
- différer
- digérer
- dilater
- diluer

- diminuer
- diplômer
- dire
- diriger
- discerner
- discréditer
- discuter
- disparaître
- dispenser
- disperser
- disposer
- disproportionner
- disputer
- dissimuler
- dissiper
- dissocier
- dissoudre
- dissuader
- disséminer
- distiller
- distinguer
- distraire
- distribuer
- diversifier
- divertir
- diviser
- divorcer
- divulguer
- documenter
- domicilier
- dominer
- donner
- doper
- dorer
- dormir
- doser
- doter
- doubler
- douer
- douter
- drainer
- dresser
- durcir
- durer
- dynamiser
- débarquer
- débarrasser
- débattre
- débiter
- débloquer
- déborder
- déboucher
- débourser
- débrancher
- débrouiller
- débuter
- décaler
- déceler
- décentraliser
- décerner
- décevoir
- décharger
- déchaîner
- déchiffrer
- déchirer
- déchoir
- décider
- déclarer
- déclencher
- décliner
- décoller
- décomposer
- déconnecter
- déconseiller
- décontracter
- décorer
- décortiquer
- découler
- découper
- décourager
- découvrir
- décrire
- décrocher
- décrypter
- décréter
- décéder
- dédier
- déduire
- défaire
- défavoriser
- défendre
- défier
- défiler
- définir
- défoncer
- déformer
- dégager
- dégrader
- déguiser
- déguster
- dégénérer
- déjeuner
- déjouer
- délaisser
- délibérer

- délimiter
- délivrer
- déléguer
- démanteler
- démarquer
- démarrer
- dématérialiser
- démentir
- démissionner
- démolir
- démonter
- démontrer
- démunir
- déménager
- dénaturer
- dénicher
- dénombrer
- dénommer
- dénoncer
- dénuer
- dépasser
- dépeindre
- dépendre
- dépenser
- déplacer
- déplaire
- déplorer
- déployer
- déporter
- déposer
- dépouiller
- déprimer
- déprécier
- députer
- dépêcher
- déranger
- dériver
- dérober
- déroger
- dérouler
- désactiver
- désarmer
- déserter
- désespérer
- déshabiller
- désigner
- désintéresser
- désirer
- désoler
- déstabiliser
- détacher
- détailler
- détecter
- détendre
- détenir
- déterminer
- détester
- détourner
- détruire
- détériorer
- dévaster
- développer
- déverser
- dévier
- dévoiler
- dévorer
- dévouer
- dîner
- effacer
- effectuer
- effondrer
- efforcer
- effrayer
- emballer
- embarquer
- embarrasser
- embaucher
- embellir
- embrasser
- embêter
- emmener
- emparer
- emplir
- employer
- empoisonner
- emporter
- empreindre
- empresser
- emprisonner
- emprunter
- empêcher
- encadrer
- encaisser
- enceindre
- encercler
- enchanter
- enchaîner
- enclencher
- encombrer
- encourager
- encourir
- enculer
- endetter
- endommager
- endormir

- endosser
- enduire
- endurer
- enfermer
- enfiler
- enflammer
- enfoncer
- enfouir
- enfourner
- enfuir
- engager
- engendrer
- englober
- engloutir
- enlever
- enneiger
- ennuyer
- enquêter
- enraciner
- enrayer
- enregistrer
- enrichir
- enrouler
- enseigner
- ensoleiller
- ensuivre
- entacher
- entamer
- entasser
- entendre
- enterrer
- entourer
- entraver
- entraîner
- entreposer
- entreprendre
- entrer
- entretenir
- entrevoir
- entériner
- envahir
- envelopper
- envier
- envisager
- envoler
- envoyer
- errer
- escompter
- espacer
- espérer
- esquisser
- essayer
- essuyer
- ester
- estimer
- estomper
- exacerber
- exagérer
- exalter
- examiner
- exceller
- excepter
- exciter
- exclamer
- exclure
- excuser
- excéder
- exercer
- exhiber
- exhorter
- exiger
- exiler
- exister
- exonérer
- expirer
- expliciter
- expliquer
- exploiter
- explorer
- exploser
- exporter
- exposer
- exprimer
- expulser
- expédier
- expérimenter
- exterminer
- extraire
- exécuter
- fabriquer
- faciliter
- facturer
- faillir
- faire
- falloir
- familiariser
- farcir
- fasciner
- fatiguer
- fausser
- favoriser
- façonner
- fendre
- fermer
- ferrer

- feuilleter
- fiche
- fidéliser
- fier
- figer
- figurer
- filer
- filmer
- filtrer
- finaliser
- financer
- finir
- fixer
- flanquer
- flatter
- fleurir
- flotter
- focaliser
- foncer
- fonctionner
- fonder
- fondre
- forcer
- forer
- forfaire
- forger
- formaliser
- formater
- former
- formuler
- fortifier
- fouetter
- fouiller
- fouler
- fournir
- fourrer
- foutre
- fracturer
- fragiliser
- franchir
- frapper
- freiner
- frire
- frotter
- frustrer
- fréquenter
- frôler
- fuir
- fumer
- fuser
- fusiller
- fusionner
- fâcher
- fédérer
- féliciter
- fêter
- gager
- gagner
- garantir
- garder
- garer
- garnir
- gazer
- geler
- germer
- glacer
- glisser
- glorifier
- goder
- gommer
- gonfler
- gouverner
- goûter
- grandir
- granuler
- gratter
- graver
- gravir
- greffer
- griller
- grimper
- grossir
- grouper
- guetter
- guider
- guérir
- gâcher
- gâter
- gémir
- généraliser
- générer
- gérer
- gésir
- gêner
- habiliter
- habiller
- habiter
- habituer
- hacher
- handicaper
- hanter
- harceler
- harmoniser
- hausser
- haver

- haïr
- heurter
- hisser
- hiérarchiser
- homogénéiser
- homologuer
- honorer
- hospitaliser
- huer
- humilier
- hurler
- hydrater
- hâter
- héberger
- hériter
- hésiter
- identifier
- ignorer
- illimiter
- illuminer
- illustrer
- imaginer
- imiter
- immatriculer
- immerger
- immigrer
- immobiliser
- immortaliser
- impacter
- implanter
- impliquer
- implémenter
- importer
- imposer
- impressionner
- imprimer
- improviser
- imprégner
- impulser
- imputer
- inaugurer
- incarcérer
- incarner
- incendier
- inciter
- incliner
- inclure
- incomber
- incorporer
- incriminer
- incruster
- indemniser
- indexer
- indigner
- indiquer
- individualiser
- induire
- industrialiser
- infecter
- infiltrer
- infliger
- influencer
- influer
- informatiser
- informer
- inhumer
- initier
- injecter
- innover
- inonder
- inquiéter
- inscrire
- insister
- inspirer
- installer
- instaurer
- instituer
- instruire
- insulter
- insurger
- insérer
- intensifier
- interagir
- intercepter
- interdire
- interner
- interpeller
- interpréter
- interroger
- interrompre
- intervenir
- interviewer
- intituler
- intriguer
- introduire
- intégrer
- intéresser
- inventer
- inverser
- investir
- inviter
- invoquer
- irriguer
- irriter
- isoler

- jaillir
- jalonner
- jeter
- joindre
- jouer
- jouir
- juger
- jurer
- justifier
- labelliser
- laisser
- lancer
- laquer
- lasser
- laver
- lever
- libérer
- licencier
- lier
- limer
- limiter
- limoger
- liquider
- lire
- lisser
- lister
- livrer
- localiser
- loger
- longer
- louer
- louper
- lover
- luire
- lutter
- lâcher
- lécher
- légitimer
- léguer
- maigrir
- maintenir
- majorer
- malter
- maltraiter
- manager
- mandater
- manger
- manier
- manifester
- manipuler
- manquer
- maquiller
- marcher
- marger
- marier
- mariner
- marquer
- marrer
- marteler
- masquer
- massacrer
- masser
- masturber
- mater
- matérialiser
- maudire
- maximiser
- maîtriser
- menacer
- mener
- mentionner
- mentir
- mesurer
- mettre
- meubler
- migrer
- militer
- millésimer
- miner
- minimiser
- miser
- mitiger
- mixer
- mobiliser
- modeler
- moderniser
- modifier
- moduler
- modéliser
- modérer
- moisir
- monopoliser
- monter
- montrer
- moquer
- mordre
- motiver
- motoriser
- mouiller
- mouler
- mourir
- mousser
- mouvementer
- mouvoir
- moyenner

- multiplier
- munir
- murer
- murmurer
- muscler
- muser
- muter
- mutualiser
- méconnaître
- médailler
- médiatiser
- méditer
- méfier
- mélanger
- mémoriser
- ménager
- mépriser
- mériter
- mêler
- mûrir
- nager
- naviguer
- naître
- nettoyer
- neutraliser
- nicher
- nier
- nommer
- normaliser
- noter
- notifier
- nouer
- nourrir
- noyer
- nuancer
- nuire
- numériser
- numéroter
- nécessiter
- négliger
- négocier
- obliger
- obscurcir
- observer
- obstiner
- obséder
- obtenir
- obéir
- occasionner
- occulter
- occuper
- octroyer
- oeuvrer
- officialiser
- officier
- offrir
- ombrager
- omettre
- onduler
- opposer
- opter
- optimiser
- opérer
- oranger
- orchestrer
- ordonner
- organiser
- orienter
- orner
- osciller
- oser
- oublier
- outrer
- ouvrer
- ouvrir
- ouïr
- pager
- pallier
- palmer
- paner
- panoramiquer
- paralyser
- paramétrer
- paraître
- parcourir
- pardonner
- parer
- parfumer
- parier
- parler
- parrainer
- parsemer
- partager
- participer
- partir
- parvenir
- passer
- passionner
- patienter
- paver
- payer
- peindre
- peiner
- peler
- pencher

- pendre
- penser
- percer
- percevoir
- percher
- percuter
- perdre
- perdurer
- perfectionner
- permettre
- permuter
- perpétrer
- perpétuer
- persister
- personnaliser
- persuader
- persécuter
- perturber
- peser
- peupler
- photographier
- piller
- piloter
- pincer
- piocher
- piquer
- pirater
- piéger
- placer
- plafonner
- plaider
- plaindre
- plaire
- plaisanter
- planer
- planifier
- planter
- plaquer
- pleurer
- pleuvoir
- plier
- plomber
- plonger
- plébisciter
- pointer
- poivrer
- polir
- polluer
- pomper
- ponctuer
- pondre
- populariser
- porter
- poser
- positionner
- posséder
- poster
- postuler
- pourrir
- poursuivre
- pourvoir
- pousser
- pouvoir
- pratiquer
- prendre
- prescrire
- pressentir
- presser
- prier
- primer
- priser
- privatiser
- priver
- privilégier
- proclamer
- procurer
- procéder
- prodiguer
- produire
- profiler
- profiter
- programmer
- progresser
- projeter
- prolonger
- promener
- promettre
- promouvoir
- promulguer
- prononcer
- propager
- proposer
- propulser
- proscrire
- prospérer
- prostituer
- protester
- protéger
- prouver
- provenir
- provoquer
- précipiter
- préciser
- préciter
- préconiser

- précéder
- prédire
- préférer
- prélever
- préoccuper
- préparer
- présenter
- préserver
- présider
- présumer
- prétendre
- prévaloir
- prévenir
- prévoir
- prêcher
- prêter
- prôner
- publier
- puiser
- pulvériser
- punir
- purger
- purifier
- pécher
- pénaliser
- pénétrer
- pérenniser
- périr
- péter
- pêcher
- qualifier
- quantifier
- questionner
- quitter
- rabattre
- raccorder
- raccourcir
- racheter
- raconter
- raffiner
- rafraîchir
- raisonner
- rajouter
- ralentir
- rallier
- ramasser
- ramener
- ramer
- randonner
- ranger
- rapatrier
- rappeler
- rapporter
- rapprocher
- raser
- rassembler
- rassurer
- rater
- ratifier
- rattacher
- rattraper
- ravager
- ravir
- raviver
- rayer
- rayonner
- rebaptiser
- rebondir
- receler
- recenser
- recentrer
- recevoir
- recharger
- rechercher
- recommander
- recommencer
- reconduire
- reconnaître
- reconquérir
- reconstituer
- reconstruire
- reconvertir
- recopier
- recouper
- recourir
- recouvrer
- recouvrir
- recruter
- recréer
- rectifier
- recueillir
- reculer
- recycler
- redescendre
- redevenir
- redire
- rediriger
- redistribuer
- redonner
- redoubler
- redouter
- redresser
- redécouvrir
- redéfinir
- redémarrer

- refaire
- refermer
- refléter
- refouler
- refroidir
- refuser
- regagner
- regarder
- regorger
- regretter
- regrouper
- rehausser
- rejeter
- rejoindre
- rejouer
- relancer
- relater
- relativiser
- relaxer
- relayer
- relever
- relier
- relire
- relâcher
- reléguer
- remanier
- remarquer
- rembourser
- remercier
- remettre
- remonter
- remplacer
- remplir
- remporter
- remuer
- remédier
- renaître
- rencontrer
- rendre
- renfermer
- renforcer
- renier
- renommer
- renoncer
- renouer
- renouveler
- renseigner
- rentrer
- renverser
- renvoyer
- reparler
- repartir
- repasser
- repenser
- repentir
- replacer
- replier
- replonger
- reporter
- reposer
- repousser
- reprendre
- reprocher
- reproduire
- représenter
- repérer
- requérir
- respecter
- respirer
- ressembler
- ressentir
- resserrer
- ressortir
- ressourcer
- ressusciter
- restaurer
- rester
- restituer
- restreindre
- retarder
- retenir
- retentir
- retirer
- retomber
- retoucher
- retourner
- retracer
- retraiter
- retrancher
- retranscrire
- retransmettre
- retravailler
- retrouver
- revancher
- revendiquer
- revendre
- revenir
- reverser
- revisiter
- revivre
- revoir
- revêtir
- rider
- rigoler
- rimer

- rincer
- rire
- risquer
- rivaliser
- river
- rocher
- roder
- rompre
- ronger
- roser
- rougir
- rouler
- roussir
- rouvrir
- ruer
- ruiner
- rythmer
- râper
- réactiver
- réaffirmer
- réagir
- réaliser
- réapparaître
- récapituler
- réchauffer
- réciter
- réclamer
- récolter
- récompenser
- réconcilier
- récupérer
- rédiger
- réduire
- réfléchir
- réformer
- réfugier
- réfuter
- référencer
- référer
- régaler
- régir
- réglementer
- régler
- régner
- réguler
- régénérer
- réhabiliter
- réinstaller
- réintégrer
- réinventer
- réitérer
- réjouir
- rémunérer
- rénover
- réorganiser
- répandre
- réparer
- répartir
- répercuter
- répertorier
- répliquer
- répondre
- réprimer
- réputer
- répéter
- réserver
- résider
- résigner
- résilier
- résister
- résonner
- résoudre
- résulter
- résumer
- rétablir
- rétorquer
- réunir
- réussir
- réutiliser
- réveiller
- réviser
- révolter
- révolutionner
- révoquer
- révéler
- réécrire
- rééditer
- réélire
- rêver
- rôtir
- sacrer
- sacrifier
- saigner
- saisir
- salarier
- saler
- salir
- saluer
- sanctionner
- satisfaire
- saturer
- saupoudrer
- sauter
- sauvegarder
- sauver

- savoir
- savourer
- scanner
- sceller
- scinder
- scolariser
- scruter
- sculpter
- seconder
- secouer
- secourir
- sembler
- semer
- sensibiliser
- sentir
- seoir
- serrer
- servir
- siffler
- signaler
- signer
- signifier
- sillonner
- simplifier
- simuler
- situer
- siéger
- soigner
- solder
- solliciter
- sombrer
- sommer
- sonder
- songer
- sonner
- sophistiquer
- sortir
- soucier
- souder
- souffler
- souffrir
- souhaiter
- souiller
- soulager
- soulever
- souligner
- soumettre
- soupirer
- soupçonner
- sourire
- sous-entendre
- souscrire
- soustraire
- soutenir
- souvenir
- sponsoriser
- spécialiser
- spécifier
- stabiliser
- stagner
- standardiser
- stationner
- statuer
- stigmatiser
- stimuler
- stipuler
- stocker
- stopper
- stresser
- structurer
- subir
- sublimer
- submerger
- subordonner
- subsister
- substituer
- subventionner
- succomber
- succéder
- sucer
- sucrer
- suer
- suffire
- suggérer
- suicider
- suivre
- superposer
- superviser
- supplier
- suppléer
- supporter
- supposer
- supprimer
- surcharger
- surfer
- surgir
- surmener
- surmonter
- surnommer
- surpasser
- surplomber
- surprendre
- surveiller
- survenir
- survivre

- survoler
- susciter
- suspecter
- suspendre
- symboliser
- synchroniser
- synthétiser
- sécher
- sécuriser
- séduire
- séjourner
- sélectionner
- séparer
- sévir
- tailler
- taire
- tamponner
- taper
- tapir
- tarder
- taxer
- teinter
- tempérer
- tendre
- tenir
- tenter
- terminer
- ternir
- terrasser
- terrer
- tester
- tirer
- tisser
- titrer
- tolérer
- tomber
- tomer
- tordre
- torturer
- totaliser
- toucher
- tourmenter
- tourner
- tracer
- traduire
- trahir
- traire
- traiter
- trancher
- transcender
- transcrire
- transformer
- transférer
- transiter
- transmettre
- transporter
- transposer
- traquer
- travailler
- traverser
- traîner
- trembler
- tremper
- tresser
- tricoter
- trier
- triompher
- tromper
- troubler
- trouver
- tuber
- tuer
- twitter
- typer
- tâcher
- télécharger
- téléphoner
- témoigner
- unifier
- unir
- user
- utiliser
- vacciner
- vaincre
- valider
- valoir
- valoriser
- vanter
- varier
- veiller
- vendre
- venger
- venir
- venter
- ventiler
- vernir
- verrouiller
- verser
- vibrer
- vider
- vieillir
- violer
- virer
- viser
- visionner

- visiter
- visser
- visualiser
- vitrer
- vivre
- voiler
- voir
- voler
- vomir
- voter
- vouer
- vouloir
- voyager
- véhiculer
- vénérer
- vérifier
- vêtir
- zapper
- zipper
- ébranler
- écarter
- échanger
- échapper
- échouer
- éclaircir
- éclairer
- éclater
- économiser
- écosser
- écouler
- écouter
- écraser
- écrier
- écrire
- écrouler
- édicter
- édifier
- éditer
- éduquer
- égaler
- égarer
- égoutter
- élaborer
- élancer
- élargir
- élever
- éliminer
- élire
- éloigner
- émailler
- émaner
- émerger
- émerveiller
- émettre
- émigrer
- émouvoir
- énerver
- énoncer
- énumérer
- épanouir
- épargner
- éparpiller
- épauler
- épicer
- éplucher
- épouser
- éprouver
- épuiser
- épurer
- équilibrer
- équiper
- équivaloir
- éradiquer
- ériger
- établir
- étaler
- étayer
- éteindre
- étendre
- étiqueter
- étirer
- étoffer
- étoiler
- étouffer
- étudier
- évacuer
- évader
- évaluer
- évanouir
- éveiller
- éviter
- évoluer
- évoquer
- être
- ôter