Table des matières

Articles longs

Fouille de données et TAL

Sémantique

Parsing 1

Lexique 1

Gestion des erreurs en TAL

Modèles linguistiques

Méthodes numériques pour le TAL

Lexique 2

Étiquetage 1

Traitement de corpus 1

Sentiments

Outils

Étiquetage 2

Langue des signes

Résumé automatique

Influence des marqueurs multi-polaires dépendant du domaine pour la fouille d'opinion au niveau du texte

Morgane Marchand[1, 2] Romaric Besançon[1] Olivier Mesnard[1] Anne Vilnat[2]
(1) CEA, LIST, Laboratoire Vision et Ingénierie des Contenus
Centre Nano-Innov Saclay, 91191 Gif-sur-Yvette Cedex
(2) LIMSI-CNRS, Université Paris-Sud, 91403 Orsay Cedex
morgane.marchand@cea.fr, romaric.besancon@cea.fr, olivier.mesnard@cea.fr, anne.vilnat@limsi.fr

Résumé. Les méthodes de détection automatique de l'opinion dans des textes s'appuient sur l'association d'une polarité d'opinion aux mots des textes, par lexique ou par apprentissage. Or, certains mots ont des polarités qui peuvent varier selon le domaine thématique du texte. Nous proposons dans cet article une étude des mots ou groupes de mots marqueurs d'opinion au niveau du texte et qui ont une polarité changeante en fonction du domaine. Les expériences, effectuées à la fois sur des corpus français et anglais, montrent que la prise en compte de ces marqueurs permet d'améliorer de manière significative la classification de l'opinion au niveau du texte lors de l'adaptation d'un domaine source à un domaine cible. Nous montrons également que ces marqueurs peuvent être utiles, de manière limitée, lorsque l'on est en présence d'un mélange de domaines. Si les domaines ne sont pas explicites, utiliser une séparation automatique des documents permet d'obtenir les mêmes améliorations.

Abstract. In this article, we propose a study on the words or multi-words which are good indicators of the opinion polarity of a text but have different polarity depending on the domain. We have performed experiments on French and English corpora, which show that taking these multi-polarity words into account improve the opinion classification at text level in a domain adaptation framework. We also show that these words are useful when the corpus contains several domains. If these domains are not explicit, using a automatic domain characterization (e.g. wich Topic Modeling approaches) allows to achieve the same results.

Mots-clés : Fouille d'opinion, adaptation au domaine, marqueurs multi-polaires.

Keywords: Opinion mining, domain adaptation, multi-polarity markers.

Introduction

Avec l'avènement du web 2.0, la manière dont les personnes expriment leur opinion a beaucoup changée : nous postons des critiques de produits de consommation sur des sites marchands et exposons nos points de vue sur presque tous les sujets, sur des forums, des groupes de discussion ou des blogs. Tout cela constitue une importante source d'information avec de nombreuses applications. C'est pourquoi, au cours des dernières années, de nombreux travaux ont pris pour objet la fouille d'opinion. Si beaucoup de ces travaux se focalisent sur la caractérisation de l'opinion sur un corpus donné, qui est souvent spécifique à un domaine, l'étude des mots qui n'indiquent pas la même opinion d'un domaine à l'autre est moins fréquente. Certains mots peuvent en effet changer de polarité entre deux domaines (Navigli, 2012; Yoshida *et al.*, 2011). Par exemple, le mot "retourner" a une connotation positive dans la phrase "Je n'en peux plus d'attendre pour retourner à mon livre !". Mais il exprime généralement une opinion très négative s'il est employé pour parler d'un appareil électronique, comme dans "J'ai dû retourner au magasin". Ce phénomène peut apparaître même lorsque les domaines sont très proches : "J'étais mort de rire" est bon signe pour un film comique mais pas pour un film d'horreur. Dans cet article, les mots ou groupes de mots sujets à ce phénomène sont appelés des "marqueurs multi-polaires". Si on ne repère pas de tels mots lors d'une tâche de classification automatique de l'opinion, ils peuvent conduire à des erreurs de classification (Wilson *et al.*, 2009). Nous proposons ici une étude de ces marqueurs, en montrant l'apport potentiel de leur prise en compte pour l'adaptation d'un domaine à un autre ainsi que pour la détection d'opinion en domaine ouvert.
Dans une première partie, nous explicitons le concept de marqueur multi-polaire et le comparons avec les autres concepts présents dans l'état de l'art. Nous présentons ensuite la méthode utilisée pour détecter ces marqueurs, ainsi qu'une clas-

sification de la nature de ces marqueurs. Dans les parties suivantes, nous étudions l'influence des ces marqueurs sur la performance des classifieurs automatiques d'opinion lors du transfert d'un domaine source à un domaine cible, ainsi que sur la détection d'opinion sur dans corpus multi-domaines et des corpus en domaine ouvert.

1 Concept et état de l'art

Subjectivité, polarité et domaines

Les expressions subjectives sont des mots ou des groupes de mots utilisés pour exprimer des états mentaux comme la spéculation, l'évaluation, le sentiment ou la conviction (Wiebe *et al.*, 2005; Wiebe & Mihalcea, 2006; Wilson, 2008; Akkaya *et al.*, 2009a). Ils sont appelés "état privés", c'est à dire que ce sont des états internes qui ne peuvent pas être directement observés par les autres (Quirk & Crystal, 1985). La polarité d'un mot ou d'un sens particulier d'un mot, au contraire, fait référence à l'opinion positive ou négative qu'a un agent sur un objet particulier. Ces deux notions ne sont bien sûr pas indépendantes et la plupart des sens subjectifs des mots ont une polarité claire. Néanmoins, une expression polarisée peut également apparaître dans un contexte neutre (Wilson *et al.*, 2009). De plus, une polarité peut être associée à des mots ou des sens de mots objectifs. (Su & Markert, 2008) donnent l'exemple du mot *tuberculose* : ce mot ne décrit pas un état privé, on peut le vérifier de manière objective et sa présence dans une phrase ne force pas cette dernière à être porteuse d'opinion. Mais pour la plupart des gens, ce mot porte tout de même une forte connotation négative. Comme (Su & Markert, 2008), nous ne considérons pas que le fait d'être polaire soit réservé aux mots ou expressions ayant été au préalable classés comme subjectifs.
La polarité d'un mot ou d'une expression peut de plus varier en fonction du contexte. Depuis quelques années, l'intérêt pour lever l'ambiguïté sur la polarité des mots ambigus s'est amplifié (Wu & Jin, 2010). Presque tous les schémas d'annotation existant pour la polarité permettent de noter cette ambiguïté (Su & Markert, 2008; Wilson *et al.*, 2005). Nous nous intéressons ici spécifiquement aux variations de polarité dues au domaine du texte, c'est à dire à son type de sujet. Dans leur travail sur la polarité contextuelle, (Wilson *et al.*, 2005) incluent le sujet et le domaine comme causes possibles de variation de polarité. De plus, (Su & Markert, 2008) remarquent dans leur étude que des préférences de polarité existent selon le domaine ou le sujet du texte. Leur corpus contient 32,5 % de mots à la polarité ambiguë et la simple désambiguïsation de sens ne parvient pas à résoudre complètement cette ambiguïté. Dans (Takamura *et al.*, 2006, 2007), les auteurs proposent une méthode utilisant un modèle avec variable latente et réseau lexical pour déterminer l'orientation de paires adjectif+nom. Ils remarquent que si l'adjectif est ambigu, la classification est plus difficile.

Les marqueurs multi-polaires au niveau du texte

Dans cette étude, nous nous intéressons aux mots ou expressions (subjectifs tout comme objectifs) qui, de manière récurrente dans un domaine particulier, sont des indicateurs de l'opinion de l'auteur sur l'objet du texte. Tout comme pour l'exemple de la tuberculose, beaucoup de mots auxquels nous nous intéressons ne vont pas avoir de polarité intrinsèque mais peuvent apparaître dans des contextes récurrents de connotation polaire pour un domaine particulier.
Ce travail est proche des concepts de polarité contextuelle ou ciblée (Wilson *et al.*, 2005, 2009; Fahrni & Klenner, 2008). (Fahrni & Klenner, 2008) se focalisent sur la détermination de la polarité ciblée des adjectifs. Un nom spécifique à un domaine est souvent modifié par un adjectif qualificatif. D'après les auteurs, les adjectifs n'ont pas de polarité *a priori* mais une polarité ciblée. Dans certains cas, un même adjectif peut changer de polarité en fonction du nom qu'il accompagne. Les auteurs utilisent Wikipédia pour la détection automatique des mots qui peuvent potentiellement être la cible d'une opinion pour un domaine donné. Une méthode de *bootstrap* est ensuite utilisée afin de déterminer la polarité ciblée des adjectifs associés à ces mots. Ils obtiennent de bons résultats mais s'intéressent uniquement aux adjectifs. (Wilson *et al.*, 2005), quant à eux, ne se restreignent pas aux adjectifs mais travaillent uniquement sur des segments de texte contenant des mots prédéterminés (des mots d'un lexique ayant au moins un sens subjectif). Ils se placent au niveau du segment et déterminent d'abord si une expression est neutre ou polaire avant de désambiguïser la polarité des expressions polaires en utilisant des règles manuelles et des traits structurels. Leur lexique couvre 75 % des segments polaires de leur corpus.
Pour notre étude, nous ne présumons pas des mots ou des expressions qui sont porteurs ou non d'information polaire. Nous avons donc choisi de les sélectionner automatiquement et de les classifier en une seule étape.
De plus, nous nous intéressons dans cet article à l'influence des mots ou expressions à polarité ambiguë (que nous appellerons marqueurs multi-polaires) sur la valeur de la polarité du texte entier. Beaucoup de travaux utilisent un lexique donnant la polarité *a priori* des mots. Ces lexiques sont souvent construits en étendant un petit lexique initial, soit en tirant

parti des conjonctions comme et/mais (Hatzivassiloglou & McKeown, 1997), soit en mesurant la co-occurrence entre mots dans un corpus ou à l'aide de moteurs de recherche (Turney & Littman, 2002). D'autres travaux améliorent un lexique déjà pré-existant, par exemple en pondérant les différentes polarités possibles d'un mot en fonction du domaine (Choi & Cardie, 2009). Ces lexiques particuliers peuvent alors être utilisés dans des classifieurs à base de règles pour classer la polarité des textes entiers (Ding *et al.*, 2008). Les études au niveau du texte utilisant des classifieurs à base de corpus s'intéressent, quant à elles, principalement à la représentation des données (Glorot *et al.*, 2011; Huang & Yates, 2012). L'erreur d'adaptation d'un classifieur dépend en effet de sa performance sur le domaine source ainsi que de la distance entre les distributions des mots dans les domaines source et cible (Ben-David *et al.*, 2007). Avec une bonne projection, un lien peut être établi entre les mots du domaine cible qui n'existent pas dans le domaine source et les autres mots (Pan *et al.*, 2010; Blitzer *et al.*, 2007). Cependant, si un mot a une polarité différente dans le domaine source et le domaine cible, cela va introduire une erreur d'adaptation. Ainsi, la détection des marqueurs multi-polaires est complémentaire à ces approches et leurs améliorations respectives peuvent être combinées.

2 Caractérisation des marqueurs multi-polaires

2.1 Méthode de détection des marqueurs multi-polaires

Nous nous intéressons donc aux marqueurs de polarité d'opinion au niveau du texte, dont la polarité est changeante avec le domaine. Le repérage automatique de ces mots ou expressions se fait par apprentissage, en utilisant des corpus de textes issus de différents domaines et annotés globalement en fonction de leur polarité sur un axe positif-négatif. Plus précisément, pour chaque mot apparaissant dans plusieurs corpus, nous regardons si sa distribution dans les critiques positives et négatives est statistiquement différente selon les domaines [1]. La caractérisation de cette différence statistique est établie par un test du χ^2 avec un risque de première espèce (i.e. risque de faux positif) de 1 %.

2.2 Utilisation des marqueurs multi-polaires lors d'un transfert

Une fois les marqueurs multi-polaires détectés, ils peuvent servir à améliorer la classification d'opinion lors d'un transfert d'un domaine source à un domaine cible. Lors de cette tâche, un classifieur d'opinion est automatiquement appris sur le corpus source annoté avant d'être utilisé sur un domaine cible. Afin de prendre en compte l'information apportée par les marqueurs d'opinion multi-polaires détectés, nous proposons de modifier les corpus source et cible avant l'entraînement du classifieur. Nous proposons deux types de modifications différentes :

En distinguant les mots Chaque marqueur multi-polaire est différencié selon le domaine : il est remplacé par le trait *marqueur_Source* dans le corpus d'entraînement (du domaine source) et par *marqueur_Cible* dans le corpus de test (du domaine cible). Ainsi, l'erreur de transfert sur les marqueurs sélectionnés est évitée.

En enlevant les mots Chaque marqueur multi-polaire est tout simplement retiré, à la fois du corpus d'entraînement et du corpus de test.

2.3 Classification des marqueurs multi-polaires

Les changements de polarité que l'on observe dans les textes, peuvent être liés à des phénomènes linguistiques ou contextuels différents. Nous en proposons la classification suivante :

Changement de sens La multi-polarité d'un mot peut être liée à sa polysémie. Dans "*I had to return my phone to the store*"ou "*I can't wait to return to my book*", le mot *return* a une polarité différente car il s'agit de deux sens différents. Dans ce cas, utiliser une méthode de désambiguïsation de sens ou de subjectivité comme dans (Akkaya *et al.*, 2009b) peut être utile.

Qualité relative Certains adjectifs ou qualificatifs sans polarité *a priori* peuvent être positifs ou négatifs en fonction de l'objet qu'ils qualifient (Fahrni & Klenner, 2008). Être "imprévisible" est un qualificatif positif pour un scénario de film mais négatif pour un logiciel.

1. Certains mots peuvent changer de polarité à l'intérieur du même domaine mais nous ne nous intéressons ici qu'à la polarité au niveau global.

Orientation morale et politique de l'auteur Certains mots peuvent changer de polarité en fonction de l'opinion de l'auteur. Cela concerne souvent les termes politiques (par exemple "capitalisme").

Comparaison Les opinions comparatives ("meilleur que...") sont difficiles à prendre en compte car il faut alors savoir quelle partie de la comparaison est l'objet principal du texte. Des travaux sont consacrés à ce problème spécifique (Ganapathibhotla & Liu, 2008). Nous avons pu détecter des habitudes générales dans nos différents corpus. Pour certains, l'objet de la critique est dans une très grande majorité à la première place de la comparaison. Dans d'autres, c'est le contraire. Il peut cependant être délicat de vérifier qu'il s'agit bien d'un phénomène global propre à un domaine et non un biais de corpus.

Aspect temporel La polarité de certains mots peut être connectée à une information temporelle. Par exemple, "*I loved this book*" est positif mais "*I loved this camera*" est habituellement négatif car l'objet ne fonctionne en général plus. Ainsi, "*I loved*" est le signe d'une opinion négative lorsque l'on parle d'objets électroniques mais la forme au présent, "*I love*", reste positive.

Biais de corpus Un changement de polarité peut être dû à un biais de corpus. Par exemple, si beaucoup de monde est d'accord pour dire que le film *Superman* est une adaptation peu réussie de la bande dessinée classique, le mot *Superman* risque d'être associé à une polarité fortement négative dans un corpus dédié aux critiques de films.

Pour certaines de ces catégories, des traitements spécifiques existent, comme la désambiguïsation de sens ou les travaux sur les opinions comparatives. Pour d'autres, il n'existe pas de traitement usuel. C'est pourquoi étudier ces mots multi-polaires est une nécessité.
Une annotation manuelle est actuellement en cours afin d'étudier la répartition des marqueurs multi-polaires dans ces différentes classes. Nous nous attacherons notamment à comparer les phénomènes observés sur l'anglais et le français afin d'expliquer plus en détail les différences observées sur les résultats.

3 Impact des marqueurs multi-polaires pour l'adaptation au domaine

3.1 Extraction des marqueurs multi-polaires

Nous avons réalisé la détection de marqueurs multipolaires pour l'anglais et le français. Pour l'anglais, nous avons utilisé les corpus *Multi-Domain Sentiment Dataset* (MDSD), collectés par (Blitzer *et al.*, 2007). Il s'agit de quatre corpus thématiques (*DVDs*, *kitchen*, *electronics* et *books*) contenant des critiques collectées sur le site internet Amazon. Chacun des corpus thématiques contient 1000 critiques positives et 1000 critiques négatives que nous utilisons pour la détection des marqueurs multi-polaires. Ces corpus contiennent également un certain nombre de critiques supplémentaires qui seront utilisées pour le test des expériences présentées dans les parties suivantes (de 3586 à 5945 selon le corpus). Les textes sont représentés en sacs de mots de bi-grammes et uni-grammes des formes fléchies des mots pleins. Leurs nombres d'occurrences sont pondérés par la taille du texte.
Pour le français, nous avons utilisé les corpus *JeuxVideo* et *AvoirAlire* issus du Défi Fouille de Textes 2007 (DEFT) (Grouin *et al.*, 2007). Ces corpus contiennent des critiques issues des sites *avoir-alire.com* et *jeuxvideo.com*. Elles sont réparties en trois classes, positif, neutre et négatif mais nous ne considérons ici que les classes positif et négatif. Comme le corpus *AvoirAlire* contient des critiques de différents domaines (films, musiques, livres, pièces de théâtre...), une séparation manuelle selon ces sous-domaines a été effectuée. Les critiques de ces corpus sont majoritairement étiquetées positif ou neutre. Seule la sous-partie *films* contient suffisamment de critiques négatives pour représenter un corpus d'apprentissage équilibré. Pour la détection des marqueurs multi-polaires, nous avons donc utilisé des corpus constitués de critiques sélectionnées au hasard dans la sous-partie *films* de *AvoirAlire* ainsi que dans *JeuxVideo* afin de constituer deux corpus thématiques équilibrés. Chacun contient 420 critiques positives et 420 critiques négatives. Le reste des critiques est utilisé pour le test lors des expériences des parties suivantes (293 textes pour *films*, 1446 pour *jeux vidéo*). Comme pour l'anglais, les textes sont représentés en sacs de mots pondérés de bi-grammes et uni-grammes des formes fléchies.

L'expérience se déroule de la manière suivante :

Détection des marqueurs multi-polaires
Pour l'extraction des marqueurs multi-polaires, nous utilisons les sous-parties annotées des domaines source et cible et réalisons le test du χ^2 comme décrit à la section 2.1. La sous-partie annotée du corpus cible n'est par contre pas utilisée pour entraîner le classifieur d'opinion. En effet, l'objectif de ce test est de valider que les mots multi-polaires ont un impact sur la détection d'opinion quand on change de domaine. Nous utilisons cette supervision pour extraire

les marqueurs multi-polaires, de façon à produire les meilleurs marqueurs effectifs. Dans un cadre réel d'adaptation au domaine, cette détection doit être faite de façon non supervisée, sans annotation dans le domaine cible.

Entraînement des classifieurs sur le corpus source modifié
Pour la classification automatique des textes en opinion positive/négative, nous avons utilisé un algorithme de boosting : *AdaBoost* dans son implémentation *BoosTexter* (Freund *et al.*, 1996; Schapire & Singer, 2000). Comme décrit à la partie 2.2, trois classifieurs d'opinion sont entraînés sur le corpus d'entraînement du domaine source : un premier classifieur de référence sans rien modifier, un classifieur entraîné en distinguant les marqueurs entre source et cible et un classifieur entraîné en supprimant tout simplement ces marqueurs de tous les corpus.

Classification du corpus cible modifié
Pour le test, nous utilisons la totalité des textes disponibles du domaine cible (la petite sous-partie ayant servie à la détection des marqueurs multi-polaires ainsi que tous les textes de test supplémentaires).

3.2 Exemples de marqueurs multi-polaires

Le tableau 1 présente quelques marqueurs détectés comme changeant de polarité entre deux domaines dans le corpus anglais MDSD. Pour chaque domaine, un mot a un score de positivité qui correspond à son nombre d'occurrences dans des critiques positives par rapport à son nombre total d'occurrences dans le domaine. Un score de 1 (resp. 0) signifie que dans ce domaine, le mot n'apparaît que dans des critiques positives (resp. négatives). Un écart de 0.5 est donc très significatif, faisant passer un mot de neutre à fortement polarisé.

	region	*I loved*	*worry*	*compare*	*return*
Domaine *electronics*	0.154	0.091	0.929	0.846	0.055
Domaine *books*	0.818	0.735	0.3	0.263	0.633

TABLE 1 – Pourcentage de présence de cinq exemples de marqueurs dans les critiques positives pour deux domaines. Le score va de 0 (très fortement négatif) à 1 (très fortement positif).

Nous avons ainsi en moyenne détecté 400 marqueurs multi-polaires sur l'anglais et 1000 sur le français. Ce décalage est vraisemblablement dû au fait que le vocabulaire français est, dans notre exemple, plus étendu que le vocabulaire anglais. Ceci s'explique d'une part parce que le corpus français considéré est de nature légèrement différente : les auteurs des textes étant des critiques de métier, ils ont vraisemblablement un vocabulaire plus riche que les auteurs des critiques du site Amazon. D'autre part, cette détection s'appuie sur les formes fléchies des mots, qui sont plus nombreuses en français du fait d'une morphologie plus riche. Notons que l'intégralité des marqueurs détectés selon cette méthode ne sera pas forcément utilisée par les classifieurs automatiques d'opinion. Il s'agit essentiellement d'indicateurs pour repérer d'éventuelles difficultés dans l'adaptation d'un domaine à un autre.

3.3 Évaluation

Les figures 1 et 2 présentent les résultats obtenus en exactitude (*accuracy*) respectivement pour le français et l'anglais. Ces résultats montrent que notre méthode donne de bons résultats sur le corpus français. En revanche, sur le corpus anglais, les résultats sont mitigés, avec des améliorations statistiquement significatives pour la moitié des paires testées et deux cas de détérioration. Il est cependant intéressant de noter que les meilleures améliorations sont observées pour les paires de corpus ayant le plus de difficulté de transfert (ceux dont l'exactitude du classifieur sans modification est déjà faible). Notre méthode étant purement statistique, elle ne fait pas intervenir, dans son fonctionnement théorique, des objets spécifiques à une langue. Par contre, elle s'appuie sur la segmentation en mots et les formes fléchies, ce qui peut être une piste pour expliquer les différences observées. Nous avons l'intention, dans des travaux futurs, de nous pencher de manière plus approfondie sur la sensibilité de notre méthode à la langue des textes.

52% des traits sélectionnés par BoosTexter sont des bi-grammes mais seul 42% des marqueurs multi-polaires utilisés le sont. Ainsi, en proportion, les marqueurs multi-polaires détectés sont plus souvent des uni-grammes même si la part de bi-grammes reste importante. De façon générale, on note qu'en moyenne, parmi les premiers mots choisis par BoosTexter en tant que classifieurs faibles (entre 700 et 800 selon les paires), 12 % se retrouvent dans notre liste de mots changeant de polarité en anglais et 10 % en français. Ainsi, près de 10 % des règles peuvent propager une erreur.
Il est également intéressant de noter que certains sens d'adaptation marchent mieux que d'autres. En effet, bien que pour

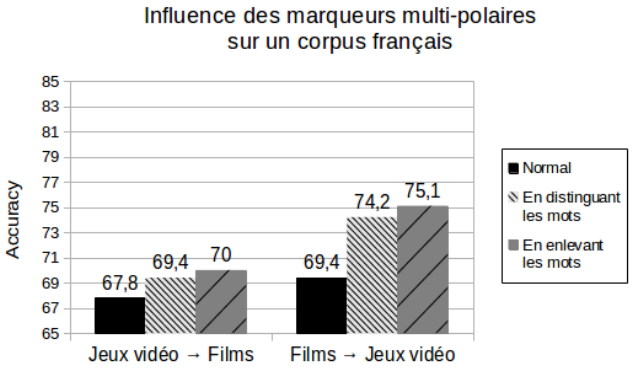


FIGURE 1 – *Accuracy* pour un classifieur entraîné sur un domaine source et testé sur un domaine cible en français (DEFT).

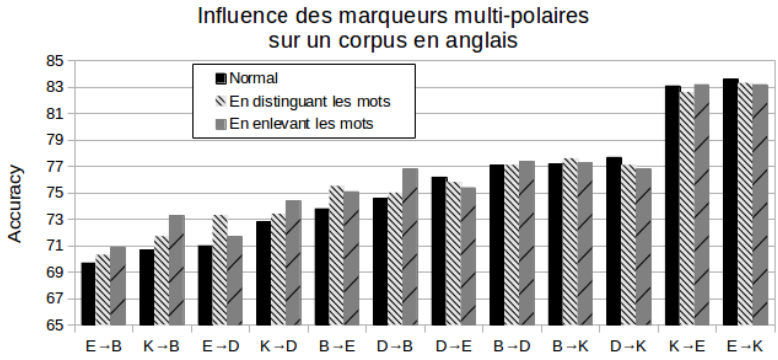


FIGURE 2 – *Accuracy* pour un classifieur entraîné sur un domaine source et testé sur un domaine cible en anglais (MDSD) ; D : DVD, B : books, E : electronics, K : kitchen.

une même paire de domaines les marqueurs changeant de polarité soient les mêmes dans un sens ou dans l'autre, ces marqueurs ne sont pas forcément utilisés comme traits dans les deux sens. Il est beaucoup plus fréquent que des marqueurs polaires deviennent neutres plutôt qu'ils passent de positif à négatif. Ainsi, si un marqueur est positif pour le domaine *films*, il sera appris comme trait positif par le classifieur. Et s'il est neutre pour le domaine *jeux vidéo*, cela provoquera une erreur de transfert. Mais dans l'autre sens, en s'entraînant sur *jeux vidéo*, rien ne sera appris pour ce trait puisqu'il est neutre. Ainsi, il n'y aura pas d'erreur de transfert bien que l'on perde de l'information. Par exemple, sur le corpus français, 92 marqueurs multi-polaires sont utilisés à l'origine dans le sens *films* vers *jeux vidéo* mais seulement 55 dans le sens *jeux vidéo* vers *films*. Aussi, il y a plus d'erreurs évitées dans un sens que dans l'autre.

Ainsi, cette prise en compte différenciée élémentaire des marqueurs changeant de polarité améliore la classification de l'opinion. Il est de plus vraisemblable qu'une pondération des marqueurs changeant de polarité, plutôt qu'une suppression complète, donne de meilleurs résultats (Choi & Cardie, 2009).

4 Utilisation des marqueurs multi-polaires pour des corpus multi-domaines

La partie précédente montre que les marqueurs multi-polaires sont utiles lors de l'adaptation d'un domaine source à un domaine cible. Nous nous plaçons à présent dans le cas où les corpus d'entraînement et de test sont chacun composés de plusieurs domaines de manière équivalente. Cela peut être le cas lorsqu'ils sont issus de la même source qui est elle-même multi-domaines, par exemple un blog abordant plusieurs sujets. Nous supposons dans cette section que la décomposition en domaines du corpus et l'attribution de chaque texte à un domaine sont connus.

4.1 Méthode

Avec un corpus multi-domaine, nous proposons une méthode de prise en compte des marqueurs multi-domaines en deux étapes : (1) la détection de marqueurs multi-polaires spécifiques à chaque domaine, (2) la construction de classifieurs d'opinions spécifiques à chaque domaine intégrant ces marqueurs.

4.1.1 Détection des marqueurs multi-polaires

Le processus de détection des marqueurs multi-polaires pour un corpus multi-domaine est présenté dans la figure 3. Le corpus d'entraînement est séparé en plusieurs sous-parties, chacune correspondant à un domaine particulier. Pour détecter les marqueurs multi-polaires, nous utilisons les étiquettes positives et négatives des données d'entraînement, comme décrit dans la section 2. Nous effectuons cette détection pour chaque sous-partie. A chaque fois, nous détectons les mots qui changent de polarité entre une sous-partie particulière du corpus d'entraînement et son complément (tous les autres textes). A la fin de cette procédure, nous avons plusieurs collections de marqueurs multi-polaires (une collection différente pour chaque sous-partie).

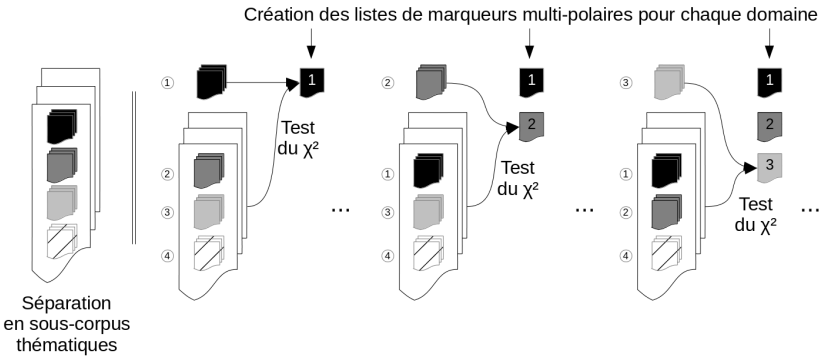


FIGURE 3 – Détection des marqueurs multi-polaires entre les sous-parties thématiques du corpus d'entraînement.

4.1.2 Différentiation des marqueurs multi-polaires

Nous créons un corpus d'entraînement différent pour chaque domaine par modification du corpus original en utilisant la liste de marqueurs multi-polaires associée à ce domaine. Pour cette expérience, nous avons testé uniquement la suppression des marqueurs multi-polaires. En effet, cette modification a donné globalement de meilleurs résultats dans nos précédentes expériences. Nous entraînons ensuite un classifieur sur ce corpus modifié et obtenons ainsi un classifieur spécifiquement adapté pour chaque domaine.

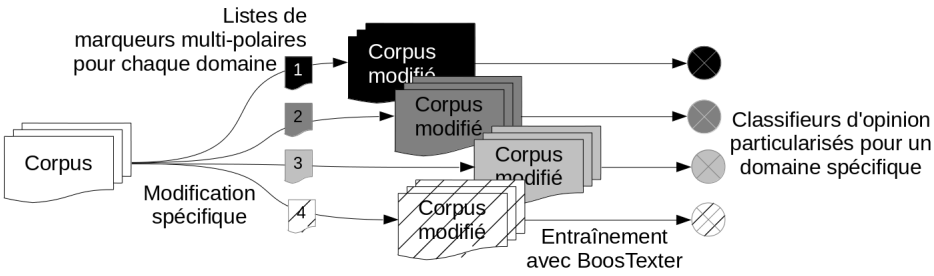


FIGURE 4 – Processus de création de plusieurs classifieurs thématiques en particularisant le corpus d'entraînement en enlevant les marqueurs multi-polaires des différentes listes associées à un domaine particulier.

Nous obtenons ainsi plusieurs classifieurs différents, chacun particularisé pour un domaine particulier (figure 4). Un texte du corpus de test est ensuite classifié en utilisant le classifieur propre à son domaine.

4.2 Évaluation des résultats

Nous avons effectué une évaluation de la méthode proposée avec le corpus *AvoirAlire* de DEFT dans son intégralité. Il y a donc 5 domaines : *livres* (757 textes), *bandes dessinées* (387 textes), *films* (1623 textes), *musique* (343 textes), *théâtre* (289 textes). Ils contiennent trois classes non équilibrées (55 % de textes positifs, 30 % de neutres et 15 % de négatifs). Pour l'anglais, nous avons de nouveau utilisé le corpus MDSD (8000 critiques annotées en positif/négatif réparties en quatre domaines : *DVDs*, *books*, *electronics* et *kitchen*).
Pour chaque corpus, nous avons réalisé une validation croisée. Le corpus est séparé aléatoirement en dix parties, neuf d'entre elles servant successivement de corpus d'entraînement et la dixième de corpus de test. Les résultats présentés sont les résultats moyens des dix expériences. Les textes sont toujours représentés en sacs de mots des uni- et bi-grammes des formes fléchies. La métrique d'évaluation utilisée lors de cette expérience est la F-mesure moyenne des classes positives et négatives.

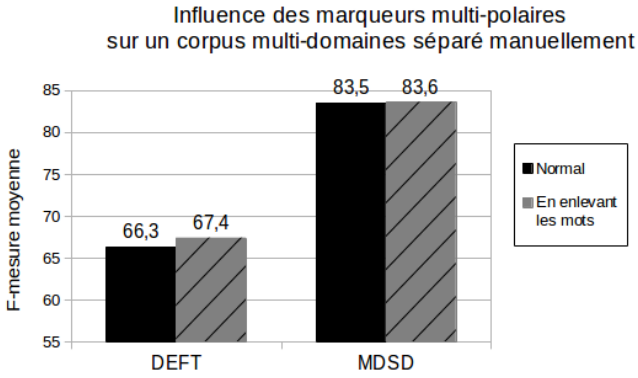


FIGURE 5 – Influence des marqueurs multi-polaires sur la performance d'un classifieur d'opinion au niveau du texte pour des corpus multi-domaines français (DEFT) et anglais (MDSD).

Les résultats, présentés à la figure 5, montrent que la prise en compte des marqueurs multi-polaires peut contribuer à améliorer la classification de l'opinion dans le cas d'un corpus contenant plusieurs domaines. Les améliorations potentielles sont cependant plus faibles que celles obtenues lors de l'adaptation d'un domaine à un autre (cf. section 3). En effet, les corpus d'entraînement et de test ont la même répartition de domaines. Notre méthode permet d'éviter des erreurs lorsqu'un mot a une certaine polarité dans tous les domaines sauf dans un où il a une polarité différente. Un apprentissage global assignera à ce mot la polarité dominante. Les erreurs ne se présenteront que dans la sous-partie du corpus de test associée avec le domaine isolé alors que, pour les autres parties du corpus de test, il n'y aura pas d'erreurs. Les erreurs que l'on peut éviter avec notre méthode sont donc moins nombreuses dans ce cas que lors de l'adaptation d'un domaine à un autre présentée dans la partie 3. Néanmoins, différencier les marqueurs multi-polaires n'est pas très difficile à mettre en place et se conjugue aisément avec les autres méthodes de classification de l'opinion en leur permettant d'éviter un certain nombre d'erreurs.

5 Utilisation des marqueurs multi-polaires pour des corpus en domaine ouvert

Dans la section précédente, nous avons fait l'hypothèse que la répartition des textes en différents domaines était connue. Or, ce n'est pas forcément le cas : certaines collections de textes contiennent des documents de différents domaines sans séparation ni indication explicite des domaines couverts. C'est par exemple le cas de corpus collectés automatiquement sur des médias particuliers, comme Twitter, qui présentent pourtant en général un grand intérêt pour des systèmes de veille d'opinion.

5.1 Méthode

La seule différence par rapport à la section précédente est l'absence d'étiquette de domaine pour les textes des corpus. Il est donc nécessaire de détecter automatiquement les différents domaines sous-jacents afin de séparer le corpus d'entraînement général en plusieurs corpus thématiques plus petits avant d'appliquer la méthode précédente. Ensuite, nous détectons les marqueurs multi-polaires et les intégrons afin de réaliser plusieurs classifieurs selon la méthode présentée dans la section 4. Néanmois, dans le cas des domaines ouverts, l'appartenance d'un texte à un domaine n'est pas une information binaire : on a en général un poids d'association entre un texte et un domaine. Pour chaque texte, les résultats des différents classifieurs spécifiques aux domaines doivent donc être fusionnés pour obtenir la classification finale de l'opinion.

5.1.1 Génération de domaines

Comme le corpus initial n'a pas d'étiquette de domaine, nous devons tout d'abord identifier les domaines sous-jacents et assigner chaque texte à un domaine. Nous avons utilisé dans ce but une méthode automatique de détection de thèmes (*Topic Models*) et, plus précisément, la méthode d'allocation de Dirichlet latente (LDA) (Blei *et al.*, 2003). Dans le cadre de la détection d'opinion, la méthode LDA a déjà été utilisée pour l'analyse de critiques focalisées sur un aspect, qui est proche de notre travail : dans (Titov & McDonald, 2008a,b), les auteurs introduisent un modèle fusionnant des *topics* locaux et globaux et utilisent les annotations manuelles des critiques afin d'améliorer l'identification des différents *topics*. D'autres travaux, tels que (Zhang *et al.*, 2013; Li *et al.*, 2010), combinent au modèle LDA des informations de sentiment ou bien des techniques de Naives Bayes afin de sortir du modèle en sac de mots.
Pour notre expérience, nous avons utilisé l'implémentation de la méthode LDA proposée dans Mallet (McCallum, 2002), qui utilise la méthode d'échantillonnage de Gibbs afin d'inférer la distribution utilisée pour la création des modèles de *topics*. Après avoir déterminé les *topics* à l'aide du corpus d'entraînement, chaque texte est représenté par un vecteur dont la taille est le nombre de *topics*, et dont chaque composante est la proportion de mots du texte qui appartient au *topic* associé à la dimension correspondante.
Le corpus d'entraînement est ensuite séparé en sous-parties, ou domaines, chacun d'entre eux associé avec l'un des *topics* sous-jacents détectés. Un texte est simplement associé au *topic* avec lequel il a le plus d'affinité. Par exemple, si sa proportion de mots appartenant à un *topic* est 55 %, il fera partie de la sous-partie du corpus associée au domaine correspondant.

5.1.2 Détection, différentiation et fusion

La détection des marqueurs multi-polaires ainsi que la différentiation du corpus d'entraînement en plusieurs corpus d'entraînement thématiques s'effectuent de la même façon qu'à la partie 4 en utilisant la partition en domaines induite par la méthode LDA. Nous obtenons ainsi plusieurs classifieurs thématiques, un par domaine.
La différence se situe lors de la classification des nouveaux textes. En effet, les textes du corpus de test n'ont pas d'étiquette de domaine. Nous devons tout d'abord déterminer leur profil de *topics* en utilisant le modèle de *topics* de la LDA. Ensuite, nous appliquons tous les classifieurs sur les nouveaux textes et obtenons plusieurs réponses différentes, une pour chaque classifieur spécifique au domaine. Nous fusionnons ces réponses en utilisant comme pondération les poids de leur profil de *topics*. Nous avons testé plusieurs stratégies de pondération pour cette fusion et la plus efficace a été de prendre l'exponentielle du score obtenu avec la LDA.

5.2 Évaluation des résultats

5.2.1 Description des corpus

Pour évaluer la méthode proposée pour les corpus en domaine ouvert, nous avons effectué tout d'abord une expérience sur les mêmes corpus français (DEFT) et anglais (MDSD) afin de pouvoir comparer avec l'expérience précédente utilisant une séparation en domaines explicite. De façon complémentaire, nous avons également utilisé le corpus anglais de tweets issu de la campagne d'évaluation SemEval 2013 pour la tâche 2 d'annotation de l'opinion (Wilson *et al.*, 2013). Ce dernier corpus est représentatif d'une collection de documents en domaine ouvert et permet de varier le type de textes sur lequel appliquer notre méthode. Les tweets sont nettoyés de leurs adresses internet, les émoticônes sont extraits et le nombre d'occurrences d'un type particulier d'émoticône (pleurs, rire, cœur...) est considéré comme un trait additionnel pour le

classifieur. Nous avons sélectionné au hasard une sous-partie équilibrée de ce corpus (1633 de chaque classe). Pour ce corpus uniquement, nous avons lemmatisé les mots du texte. En effet, les tweets étant de très courts textes, les formes fléchies ont peu d'occurrences. Comme pour les autres corpus, nous utilisons un sac de mots des uni- et bi-grammes.

La figure 6 montre que l'utilisation d'une partition automatique avec la LDA n'a pas modifié le comportement que l'on obtenait en utilisant une partition manuelle. Nous obtenons toujours une amélioration modeste sur le corpus français et des résultats similaires sur le corpus MDSD. L'utilisation d'une séparation automatique en domaines par LDA peut donc remédier à l'absence d'étiquette de domaine sans perte de performance.

5.2.2 Discussion des résultats

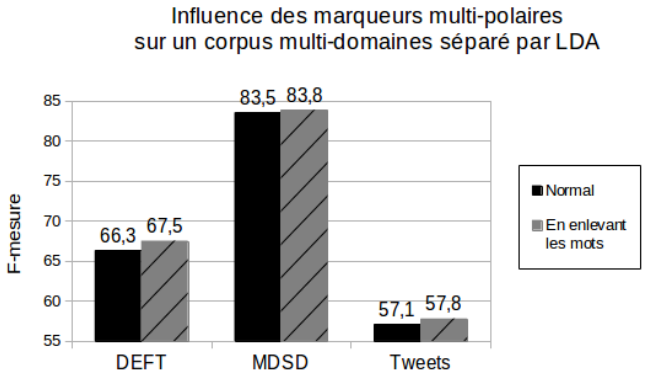


FIGURE 6 – Influence des marqueurs multi-polaires sur la performance d'un classifieur d'opinion au niveau du texte pour des corpus en domaine ouvert français (DEFT) et anglais (MDSD et Tweets). La séparation en sous-domaines thématiques a été effectuée par LDA.

Pour ce qui est du corpus de tweets, nous obtenons une très faible amélioration (+0.7 %) qui est néanmoins significative (selon un test de significativité par randomisation). Ce résultat doit être mis en relation avec le petit nombre de marqueurs multi-polaires détectés (en moyenne, 36 par domaine). Nous pensons que la taille du corpus, combinée aux 144 caractères des tweets, est trop petite pour que le test du χ_2 détecte beaucoup de marqueurs avec suffisamment de confiance. Pour comparaison, dans notre expérience sur les critiques en anglais, nous avions détecté 400 marqueurs multi-polaires par domaine. Nous nous sommes demandé si, pour ce corpus, des domaines plus focalisés sur un seul sujet pouvaient contrebalancer l'effet du manque de données.

Nous avons donc réalisé une seconde subdivision du corpus de tweets. Cette fois, un tweet n'est pris en compte que si plus de 75 % de ses mots appartiennent au même *topic*. Ainsi, un tweet dont seulement 55 % des mots appartiennent à un certain *topic* ne sera pas retenu. Dans cette version, les sous-parties du corpus d'entraînement obtenues sont plus focalisées sur un seul et même *topic*. En retour, elles contiennent moins de tweets et donc moins de données d'entraînement.

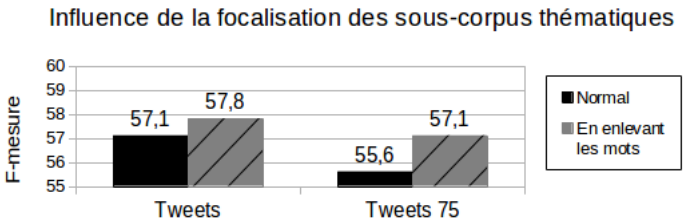


FIGURE 7 – Deux corpus d'entraînement différents sont utilisés. *Tweet* contient l'intégralité des tweets tandis que *Tweets75* contient uniquement ceux qui sont focalisés sur un seul *topic*.

La figure 7 montre le résultat de ces différentes intégrations des mots multi-polaires en utilisant deux corpus d'entraînement initiaux différents : avec l'ensemble des tweets ou avec uniquement les tweets les plus focalisés sur un *topic*. On remarque que pour l'expérience avec seulement les tweets les plus focalisés, l'amélioration est plus sensible (+1,46 % contre +0,70 %) bien que la valeur absolue du score reste inférieure en raison de la taille bien plus petite du corpus d'entraînement.

6 Conclusion

Dans cet article, nous avons étudié la notion de marqueurs multi-polaires d'opinion. Ce sont des mots ou groupes de mots qui sont indicateurs d'une certaine polarité d'opinion au niveau du texte en fonction du type d'objet dont le texte parle, ou domaine. Ces marqueurs multi-polaires sont de différents types linguistiques. Nous en avons proposé une première classification qui est actuellement en cours d'évaluation.
Nous avons testé l'apport de la prise en compte de ces marqueurs multipolaires pour la tâche de classification de l'opinion au niveau du texte lors de l'adaptation d'un domaine source à un domaine cible. Pour les corpus français étudiés, notre méthode présente une bonne amélioration de l'exactitude, allant jusqu'à +5,7 %. Pour les corpus anglais en revanche, il existe deux cas sur 12 pour lesquels la prise en compte de ces marqueurs multi-polaires dégrade significativement les performances. A l'inverse, dans 7 cas sur 12, notre méthode améliore significativement les résultats. Ces améliorations sont plus sensibles pour les paires de domaines pour lesquelles le transfert est le plus difficile.
Nous nous sommes également intéressés à l'apport possible des marqueurs multi-polaires pour la classification de l'opinion dans un corpus comportant plusieurs domaines. Nous particularisons le corpus d'entraînement pour chaque domaine et obtenons plusieurs classifieurs. Nos expériences montrent un gain moyen de +1,2 % de F-mesure pour le français. Le corpus de critiques en anglais n'obtient malheureusement pas d'augmentation significative. De plus, en l'absence de séparation explicite en domaines, le recours à un modèle de *topics* calculé par LDA ainsi qu'une fusion de classifieurs permettent d'obtenir les mêmes résultats. Nous avons également testé cette approche sur un corpus de tweets en anglais et nous trouvons une petite amélioration qui reste toutefois significative. Pour ce corpus contenant de très courts textes, nous avons montré que notre méthode est plus efficace lorsque les tweets composants le corpus d'entraînement sont focalisés précisément sur un seul domaine (+1,46 % de F-mesure).
Pour la suite de nos travaux, nous allons rechercher les phénomènes linguistiques qui influent sur la différence de performance entre français et anglais. Pour cela, nous allons poursuivre l'évaluation manuelle des marqueurs multi-polaires extraits automatiquement. Nous avons également l'intention de poursuivre nos expériences sur la façon de détecter ces marqueurs multi-polaires en utilisant le moins possible d'annotations dans le domaine cible. Une approche possible est de caractériser le comportement des mots candidats par rapport à des mots pivots de polarité stable et connue. Cela permettra de bénéficier de l'apport des marqueurs multi-polaires pour l'adaptation au domaine dans un cadre totalement non supervisé.

Références

AKKAYA C., WIEBE J. & MIHALCEA R. (2009a). Subjectivity word sense disambiguation. In *EMNLP*, p. 190–199, Singapore : Association for Computational Linguistics.

AKKAYA C., WIEBE J. & MIHALCEA R. (2009b). Subjectivity word sense disambiguation. In *EMNLP*.

BEN-DAVID S., BLITZER J., CRAMMER K. & PEREIRA F. (2007). Analysis of representations for domain adaptation. *Advances in neural information processing systems*, **19**, 137.

BLEI D. M., NG A. Y. & JORDAN M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, **3**, 993–1022.

BLITZER J., DREDZE M. & PEREIRA F. (2007). Biographies, bollywood, boom-boxes and blenders : Domain adaptation for sentiment classification. In *ACL*.

CHOI Y. & CARDIE C. (2009). Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In *EMNLP*.

DING X., LIU B. & YU P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of WSDM*, p. 231–240 : ACM.

FAHRNI A. & KLENNER M. (2008). Old wine or warm beer : Target-specific sentiment analysis of adjectives. In *Symposion on Affective Language in Human and Machine, AISB Convention*.

FREUND Y., SCHAPIRE R. E. *et al.* (1996). Experiments with a new boosting algorithm. In *ICML*, volume 96, p. 148–156.

GANAPATHIBHOTLA M. & LIU B. (2008). Mining opinions in comparative sentences. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, p. 241–248 : ACL.

GLOROT X., BORDES A. & BENGIO Y. (2011). Domain adaptation for large-scale sentiment classification : A deep learning approach. In *ICML*.

GROUIN C., BERTHELIN J.-B., EL AYARI S., HEITZ T., HURAULT-PLANTET M., JARDINO M., KHALIS Z. & LASTES M. (2007). Présentation de deft'07 (défi fouille de textes). *Actes du troisième DÉfi Fouille de Textes*, p.3̃.

HATZIVASSILOGLOU V. & MCKEOWN K. (1997). Predicting the semantic orientation of adjectives. In *EACL*, p. 174–181 : Association for Computational Linguistics.

HUANG F. & YATES A. (2012). Biased representation learning for domain adaptation. In *EMNLP*, p. 1313–1323, Jeju Island, Korea : Association for Computational Linguistics.

LI F., HUANG M. & ZHU X. (2010). Sentiment analysis with global topics and local dependency. In *Proceedings of the 24nd AAAI Conference on Artificial Intelligence (AAAI-10)*.

MCCALLUM A. K. (2002). Mallet : A machine learning for language toolkit.

NAVIGLI R. (2012). A quick tour of word sense disambiguation, induction and related approaches. *SOFSEM 2012 : Theory and Practice of Computer Science*, p. 115–129.

PAN S., NI X., SUN J., YANG Q. & CHEN Z. (2010). Cross-domain sentiment classification via spectral feature alignment. In *WWW*, p. 751–760 : ACM.

QUIRK R. & CRYSTAL D. (1985). *A comprehensive grammar of the English language*, volume 6. Cambridge Univ Press.

SCHAPIRE R. & SINGER Y. (2000). Boostexter : A boosting-based system for text categorization. *Machine learning*, **39**(2), 135–168.

SU F. & MARKERT K. (2008). From words to senses : a case study of subjectivity recognition. In *International Conference on Computational Linguistics*.

TAKAMURA H., INUI T. & OKUMURA M. (2006). Latent variable models for semantic orientations of phrases. In *EACL*.

TAKAMURA H., INUI T. & OKUMURA M. (2007). Extracting semantic orientations of phrases from dictionary. In *HLT-NAACL*, p. 292–299.

TITOV I. & MCDONALD R. (2008a). A joint model of text and aspect ratings for sentiment summarization. In *ACL*.

TITOV I. & MCDONALD R. (2008b). Modeling online reviews with multi-grain topic models. In *WWW*.

TURNEY P. & LITTMAN M. (2002). Unsupervised learning of semantic orientation from a hundred-billion-word corpus. *Arxiv preprint cs/0212012*.

WIEBE J. & MIHALCEA R. (2006). Word sense and subjectivity. In *21st International Conference on Computational Linguistics and 44th annual meeting of the Association for Computational Linguistics*, p. 1065–1072 : ACL.

WIEBE J., WILSON T. & CARDIE C. (2005). Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, **39**(2-3), 165–210.

WILSON T., KOZAREVA Z., NAKOV P., RITTER A., ROSENTHAL S. & STOYANOV V. (2013). Semeval-2013 task 2 : Sentiment analysis in twitter. In *7th International Workshop on Semantic Evaluation*.

WILSON T., WIEBE J. & HOFFMANN P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *HLT/EMNLP*.

WILSON T., WIEBE J. & HOFFMANN P. (2009). Recognizing contextual polarity : An exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, **35**, 339–433.

WILSON T. A. (2008). *Fine-grained subjectivity and sentiment analysis : recognizing the intensity, polarity, and attitudes of private states*. ProQuest.

WU Y. & JIN P. (2010). Semeval-2010 task 18 : Disambiguating sentiment ambiguous adjectives. In *5th International Workshop on Semantic Evaluation*, p. 81–85.

YOSHIDA Y., HIRAO T., IWATA T., NAGATA M. & MATSUMOTO Y. (2011). Transfer learning for multiple-domain sentiment analysis - identifying domain dependent/independent word polaritys. In *AAAI*.

ZHANG Y., JI D.-H., SU Y. & WU H. (2013). Joint naïve bayes and lda for unsupervised sentiment analysis. In *PAKDD (1)*, p. 402–413.

Influence des domaines de spécialité dans l'extraction de termes-clés

Adrien Bougouin Florian Boudin Béatrice Daille
LINA – UMR CNRS 6241, 2 rue de la Houssinière 44322 Nantes Cedex 3, France
<prenom.nom>@univ-nantes.fr

Résumé. Les termes-clés sont les mots ou les expressions polylexicales qui représentent le contenu principal d'un document. Ils sont utiles pour diverses applications, telles que l'indexation automatique ou le résumé automatique, mais ne sont pas toujours disponibles. De ce fait, nous nous intéressons à l'extraction automatique de termes-clés et, plus particulièrement, à la difficulté de cette tâche lors du traitement de documents appartenant à certaines disciplines scientifiques. Au moyen de cinq corpus représentant cinq disciplines différentes (archéologie, linguistique, sciences de l'information, psychologie et chimie), nous déduisons une échelle de difficulté disciplinaire et analysons les facteurs qui influent sur cette difficulté.

Abstract. Keyphrases are single or multi-word expressions that represent the main content of a document. Keyphrases are useful in many applications such as document indexing or text summarization. However, most documents are not provided with keyphrases. To tackle this problem, researchers propose methods to automatically extract keyphrases from documents of various nature. In this paper, we focus on the difficulty of automatic keyphrase extraction in scientific papers from various areas. Using five corpora representing five areas (archaeology, linguistics, information sciences, psychology and chemistry), we observe the difficulty scale and analyze factors inducing a higher or a lower difficulty.

Mots-clés : Extraction de termes-clés, articles scientifiques, domaines de spécialité, méthodes non-supervisées.

Keywords: Keyphrase extraction, scientific papers, specific domain, unsupervised methods.

1 Introduction

Un terme-clé est un mot ou une expression polylexicale qui représente un concept important d'un document auquel il est associé. En pratique, plusieurs termes-clés représentant des concepts différents sont associés à un même document. Ils forment alors un ensemble à partir duquel il est possible de caractériser, synthétiser, le contenu du document. Du fait de cette capacité de synthèse, les termes-clés sont utilisés dans de nombreuses applications telles que le résumé automatique (D'Avanzo & Magnini, 2005), la classification de documents (Han *et al.*, 2007) ou l'indexation automatique (Medelyan & Witten, 2008). Cependant, tous les documents ne sont pas accompagnés de termes-clés et leur assignation manuelle est une tâche coûteuse. Pour pallier ce problème, de plus en plus de chercheurs s'intéressent à l'extraction automatique de termes-clés, en témoignent les récentes campagnes d'évaluation (Paroubek *et al.*, 2012; Kim *et al.*, 2010), ainsi que les nombreux travaux à ce sujet (Hasan & Ng, 2014).

L'extraction automatique de termes-clés consiste à extraire du contenu d'un document les unités textuelles les plus importantes, celles qui permettent de le résumer. Parmi les méthodes d'extraction automatique de termes-clés existantes, nous distinguons deux catégories : les méthodes supervisées et les méthodes non-supervisées. Dans le cadre supervisé, la tâche d'extraction de termes-clés est considérée comme une tâche de classification (Witten *et al.*, 1999) où il s'agit d'attribuer la classe « *terme-clé* » ou « *non terme-clé* » aux termes-clés candidats du document. Une collection de documents annotés en termes-clés est utilisée pour l'apprentissage d'un modèle de classification reposant sur divers traits tels que la fréquence du terme-clé candidat ou sa position dans le document. Dans le cadre non-supervisé, les méthodes attribuent un score d'importance aux candidats selon divers indicateurs tels que leur degré de spécificité (Spärck Jones, 1972) ou les relations de coocurrence que leurs mots entretiennent (Mihalcea & Tarau, 2004). En général, les méthodes supervisées sont plus performantes que les méthodes non-supervisées, mais leur besoin en données d'apprentissage annotées et leur dépendance vis-à-vis du domaine de ces données d'apprentissage poussent les chercheurs à s'intéresser aux méthodes non-supervisées.

Dans cet article, nous nous plaçons dans le contexte de l'extraction non-supervisée de termes-clés à partir de documents de

nature scientifique. Faisant l'hypothèse que certaines disciplines sont plus difficiles à traiter que d'autres, nous présentons diverses stratégies d'extraction de termes-clés puis comparons leurs différences de performance. Nous déterminons ensuite quels sont les facteurs qui influent sur la difficulté de la tâche d'extraction automatique de termes-clés. De la connaissance de ces facteurs peut émerger le besoin d'utiliser des ressources externes, telles que des thésaurus, souvent mises de côté dans les travaux portant sur l'extraction non-supervisée de termes-clés. Cela peut aussi permettre de détecter la difficulté en amont de l'extraction de termes-clés afin d'affiner le paramétrage de la méthode utilisée.

Le reste de cet article est organisé comme suit. Dans un premier, temps nous présentons les collections de données (section 2) et les méthodes d'extraction de termes-clés (section 3) que nous utilisons. Dans un second temps, nous appliquons ces méthodes à nos collections de données (section 4), puis nous discutons des différents facteurs observables (section 5) avant de conclure (section 6).

2 Collections de données

Pour ce travail, nous disposons de cinq corpus disciplinaire de notices bibliographiques fournies par l'Inist [1] dans le cadre du projet ANR Termith [2] : archéologie, linguistique, sciences de l'information, psychologie et chimie. Chaque notice contient le titre, le résumé et les termes-clés d'un document auquel elle est associée. Les termes-clés sont classés en deux catégories :

- les termes-clés d'auteurs, assignés librement par les auteurs pour caractériser leur production ;
- les termes-clés Inist (en français, en anglais ou en espagnol), assignés par des indexeurs professionnels selon des règles précises destinées à améliorer la recherche d'information et à homogénéiser l'indexation des notices :
  - les termes-clés doivent être du même niveau de spécificité que celui du document et peuvent parfois être accompagnés d'un terme-clé plus générique pour le restituer dans son contexte ;
  - les termes-clés doivent respecter, autant que possible, le langage de la discipline à laquelle appartient le document (termes-clés contrôlés) ;
  - pour tous les documents d'une même discipline, un même concept doit être représenté par le même terme-clé ;
  - les termes-clés d'un document doivent présenter tous les concepts qui y sont importants, même ceux qui sont implicites.

Nous utilisons les termes-clés français assignés par l'Inist.

Le corpus d'**archéologie** est composé de 718 notices. Celles-ci représentent des articles parus entre 2001 et 2012 dans 22 revues différentes (*Paléo*, *Le bulletin de la Société préhistorique française*, etc.).

Le corpus de **linguistique** est constitué de 716 notices d'articles parus entre 2000 à 2012 dans 12 revues différentes (*Linx – Revue des linguistes de l'Université Paris Ouest Nanterre La Défense*, *Travaux de linguistique*, etc.).

Le corpus de **sciences de l'information** contient 706 notices d'articles publiés entre 2001 et 2012 dans six revues différentes (*Documentaliste – Sciences de l'information*, *Document numérique*, etc.).

Le corpus de **psychologie** contient 720 notices d'articles parus entre 2001 et 2012 dans sept revues différentes (*Enfance*, *Revue internationale de psychologie et de gestion des comportements organisationnels*, etc.).

Le corpus de **chimie** est composé de 782 notices d'articles publiés entre 1983 et 2012 dans quatre revues (*Comptes Rendus de l'Académie des Sciences*, *Comptes Rendus Chimie*, etc.).

Le tableau 1 présente les caractéristiques des cinq collections de données dont nous disposons. Les notices sont de petite taille et sont rédigées différemment selon les disciplines (cf. figure 1). Les notices d'archéologie, par exemple, font l'objet d'un effort de présentation du contexte historique lié aux travaux présentés, tandis que les notices de chimie, principalement des comptes rendus d'expériences, décrivent sommairement (énumèrent) les expériences réalisées (noms des expériences, éléments chimiques impliqués, etc.). Les termes-clés associés aux documents varient en nombre (de 8,5 à 16,6) et en complexité. Par exemple, en archéologie, nous observons qu'un grand nombre de termes-clés sont des entités nommées principalement composées d'un seul mot (p. ex. « Paléolithique », « Europe », etc.), tandis qu'en chimie, nous observons un usage fréquent de notions centrales (dans le langage de chimie) nécessitant une spécialisation systématique (p. ex. « réaction topotactique », « réaction sonochimique », « réaction électrochimique », etc.). Nous remarquons aussi

1. Institut de l'Information Scientifique et Technique : `http://www.inist.fr`
2. TERMinologie et Indexation de Textes en sciences Humaines : `http://www.atilf.fr/ressources/termith/`

Statistique	Archéologie	Linguistique	Sciences de l'information	Psychologie	Chimie
Documents	718	715	706	720	782
Mots/doc.	219,1	156,7	119,7	185,7	105,2
Termes-clés/doc.	16,6	8,0	8,5	11,6	12,8
Mots/terme-clé	1,3	1,8	1,7	1,6	2,2
Diversité des termes-clés	25,5 %	23,0 %	25,0 %	17,4 %	40,6 %
Termes-clés contrôlés	79,8 %	86,9%	85,8 %	90,9%	83,0 %
Termes-clés non contrôlés	20,2 %	13,1%	14,2 %	9,1%	17,0 %
Termes-clés extractibles (Rappel max.)	62,9 %	38,8 %	32,4 %	27,1 %	23,7 %
↪ Termes-clés contrôlés extractibles	48,8 %	34,9 %	27,9 %	24,9 %	21,7 %
↪ Termes-clés non contrôlés extractibles	14,1 %	3,9 %	4,5 %	2,2 %	2,0 %

TABLE 1 – Caractéristiques des corpus disciplinaires. La diversité des termes-clés représente la proportion de termes-clés différents dans la discipline $\left(\frac{\text{nombre de termes-clés différents}}{\text{nombre total de termes-clés}}\right)$. Les termes-clés extractibles sont les termes-clés pouvant être extraits du contenu des documents. Conformément au processus d'évaluation standard pour les méthodes d'extraction automatique de termes-clés (cf. section 4.1), les variantes flexionnelles d'un terme-clé de référence sont jugées correctes (p. ex. « langues de spécialité » peut être extrait à la place de « langue de spécialité »).

une diversité variable selon les disciplines (de 23,0 % à 40,6 % de termes-clés différents). En chimie, la diversité plus importante que pour les autres disciplines, c'est-à-dire un nombre plus important de termes-clés différents parmis tous les termes-clés de référence, indique une difficulté a priori plus importante. Enfin, il est important de noter la faible proportion de termes-clés apparaissant dans les notices, à une flexion près — rappel maximum pouvant être obtenu. Par exemple, dans le corpus de chimie, uniquement trois termes-clés peuvent être extraits des notices parmi les 12,8 associés aux notices, en moyenne, en comparant les candidats à partir de la racine de leurs mots déterminées avec la méthode de Porter (1980). Ce dernier point concerne principalement les termes-clés contrôlés, qui peuvent être assignés à un document à partir de règles concernant les unités textuelles présentent dans le document. Ces règles, dites de déclenchement, sont définies manuellement par les indexeurs professionnels et ne sont pas disponibles pour ce travail.

3 Extraction automatique de termes-clés

L'extraction non-supervisée de termes-clés peut se décomposer en quatre étapes (cf. figure 2). Tout d'abord, les documents sont un à un enrichis linguistiquement (segmentés en phrases, segmentés en mots et étiquetés en parties du discours), des termes-clés candidats y sont ensuite sélectionnés, puis ordonnés par importance et enfin, les k plus importants sont sélectionnés en tant que termes-clés. Les étapes les plus importantes d'un système d'extraction automatique de termes-clés sont celles de sélection des candidats et d'ordonnancement de ceux-ci. Intuitivement, l'ordonnancement des candidats est le cœur du système, mais la performance de celui-ci est limitée par la qualité de l'ensemble de termes-clés candidats qui lui est fourni. Un ensemble de candidats est de bonne qualité lorsqu'il fournit un maximum de candidats présents dans l'ensemble des termes-clés de référence et lorsqu'il fournit peu de candidats non-pertinents, c'est-à-dire des candidats qui ne sont pas dans l'ensemble des termes-clés de référence et qui peuvent dégrader la performance du système d'extraction de termes-clés utilisé.

3.1 Préparation des données

Les documents des collections de données utilisées subissent tous les mêmes prétraitements. Ils sont tout d'abord segmentés en phrases, puis en mots et enfin étiquetés en parties du discours. Dans ce travail, la segmentation en phrase est effectuée par le *PunktSentenceTokenizer* disponible avec la librairie Python NLTK (Bird *et al.*, 2009, *Natural Language ToolKit*), la segmentation en mots est effectuée par l'outil Bonsai du Bonsai PCFG-LA parser [3] et l'étiquetage en parties du discours est réalisé par MElt (Denis & Sagot, 2009). Tous ces outils sont utilisés avec leurs paramètres par défaut.

3. http://alpage.inria.fr/statgram/frdep/fr_stat_dep_parsing.html

Variabilité du gravettien de Kostienki (bassin moyen du Don) et des territoires associés [a] *Archéologie*

Dans la région de Kostienki-Borschevo, on observe l'expression, à ce jour, la plus orientale du modèle européen de l'évolution du Paléolithique supérieur. Elle est différente à la fois du modèle Sibérien et du modèle de l'Asie centrale. Comme ailleurs en Europe, le Gravettien apparaît à Kostienki vers 28 ka (Kostienki 8 /II/). Par la suite, entre 24-20 ka, les techno-complexes gravettiens sont représentés au moins par quatre faciès dont deux, ceux de Kostienki 21/III/ et Kostienki 4 /II/, ressemblent au Gravettien occidental et deux autres, Kostienki-Avdeevo et Kostienki 11/II/, sont des faciès propres à l'Europe de l'Est, sans analogie à l'Ouest.

Termes-clés de référence : Europe*, Kostienko, Borschevo, variation*, typologie*, industrie osseuse*, industrie lithique*, Europe centrale*, Avdeevo*, Paléolithique supérieur*, Gravettien*.

Termes techniques et marqueurs d'argumentation : pour débusquer l'argumentation cachée dans les articles de recherche [b] *Linguistique*

Les articles de recherche présentent les résultats d'une expérience qui modifie l'état de la connaissance dans le domaine concerné. Le lecteur néophyte a tendance à considérer qu'il s'agit d'une simple description et à passer à côté de l'argumentation au cours de laquelle le scientifique cherche à convaincre ses pairs de l'innovation et de l'originalité présentées dans l'article et du bien-fondé de sa démarche tout en respectant la tradition scientifique dans laquelle il s'insère. Ces propriétés spécifiques du discours scientifique peuvent s'avérer un obstacle supplémentaire à la compréhension, surtout lorsqu'il s'agit d'un article en langue étrangère. C'est pourquoi il peut être utile d'incorporer dans l'enseignement des langues de spécialité une sensibilisation aux marqueurs linguistiques (terminologiques et argumentatifs), qui permettent de dépister le développement de cette rhétorique. Les auteurs s'appuient sur deux articles dans le domaine de la microbiologie.

Termes-clés de référence : Langue scientifique*, argumentation*, rhétorique*, langue de spécialité*, enseignement des langues*, linguistique appliquée*, discours scientifique*, article de recherche.

Etude d'un condensat acide isocyanurique-urée-formaldéhyde [c] *Chimie*

La synthèse d'un condensat acide isocyanurique-urée-formaldéhyde utilisant la pyridine en tant que solvant a été effectuée par réaction sonochimique.

Termes-clés de référence : Réaction sonochimique*, hétérocycle azote*, cycle 6 chaînons*, ether*.

a. http://cat.inist.fr/?aModele=afficheN&cpsidt=20563716
b. http://cat.inist.fr/?aModele=afficheN&cpsidt=17395748
c. http://cat.inist.fr/?aModele=afficheN&cpsidt=6719275

FIGURE 1 – Exemples de notices Inist. Les termes-clés soulignés sont ceux qui occurrent dans le titre ou le résumé de la notice. Les termes-clés marqués d'une * font partie des termes-clés contrôlés.

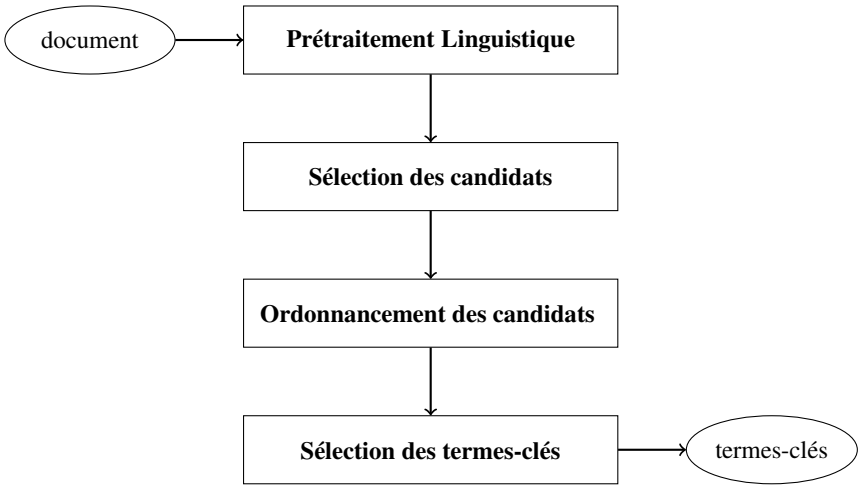


FIGURE 2 – Chaîne de traitements d'un système non-supervisé d'extraction automatique de termes-clés.

3.2 Sélection des termes-clés candidats

Dans les travaux précédents, deux approches sont fréquemment utilisées. Soit les méthodes sélectionnent les n-grammes (filtrés) en tant que termes-clés candidats, soit elles sélectionnent les candidats par reconnaissance de forme (Hulth, 2003). Dans ce travail, nous expérimentons trois méthodes différentes : deux méthodes conformes aux approches standards et une méthode sélectionnant les candidats termes obtenus par un extracteur terminologique. Aucun travail portant sur l'extraction automatique de termes-clés n'a, à notre connaissance, utilisé une telle approche. Compte tenu de la nature (disciplinaire) de nos données, nous faisons l'hypothèse que les candidats termes, tels que définis dans le domaine de l'extraction terminologique, peuvent aussi être des termes-clés candidats. Ces trois méthodes de sélection fournissent des ensembles de candidats de qualités différentes, ce qui nous permet par la suite d'identifier les facteurs qui influent sur la difficulté de l'extraction automatique de termes-clés.

La **sélection des n-grammes filtrés** consiste à extraire du document toutes les séquences ordonnées de n mots, puis à les filtrer avec un anti-dictionnaire regroupant les mots fonctionnels de la langue (conjonctions, prépositions, etc.) et les mots courants (« près », « beaucoup », etc.). Dans ce travail, nous suivons Witten *et al.* (1999) et sélectionnons les n-grammes de taille $n \in \{1..3\}$ ($\{1..3\}$-grammes) lorsque leurs mots en tête et en queue ne sont pas présents dans l'anti-dictionnaire fourni par l'université de Neuchâtel[4] (*IR Multilingual Resources at UniNE*). La sélection des n-grammes est très exhaustive, elle fournit un grand nombre de termes-clés candidats, ce qui permet de maximiser la quantité de candidats présents dans l'ensemble des termes-clés de référence, mais ce qui maximise aussi la quantité de candidats erronés (bruités).

Exemples de $\{1..3\}$-grammes sélectionnés à partir de « bassin moyen du Don » dans la notice d'archéologie de la figure 1 : « bassin », « moyen », « Don », « bassin moyen » et « moyen du Don ».

La **reconnaissance de formes** consiste à sélectionner les unités textuelles qui respectent certains patrons grammaticaux. Les termes-clés candidats sélectionnés par reconnaissance de forme ont l'avantage d'avoir une nature contrôlée avec précision (p. ex. des groupes nominaux), ce qui les rend plus fondés linguistiquement, ainsi que de meilleure qualité que les n-grammes. Dans ce travail, nous utilisons le patron `/(NOM | ADJ)+/` afin de sélectionner les plus longues séquences de noms (noms propres inclus) et d'adjectifs (Hasan & Ng, 2010).

Exemples de `/(NOM | ADJ)+/` sélectionnés à partir de « bassin moyen du Don » dans la notice d'archéologie de la figure 1 : « bassin moyen » et « Don ».

La **sélection de candidats termes** consiste à sélectionner les unités textuelles qui sont potentiellement des termes, tels que définis dans le domaine de l'extraction terminologique. En terminologie, un terme est un mot ou une séquence de mots représentant un concept spécifique à un domaine, ou une discipline. Dans ce travail, nous utilisons l'extracteur terminologique TermSuite (Rocheteau & Daille, 2011), qui est capable de détecter des candidats termes (simples et complexes) ainsi que leurs variantes. Une terminologie est extraite par TermSuite pour chaque corpus (32 119 candidats termes en

4. http://members.unine.ch/jacques.savoy/clef/index.html

Archéologie, 16 557 candidats termes en Sciences de l'Information, 21 330 candidats termes en Linguistique, 24 680 candidats termes en Psychologie et 21 020 candidats termes en Chimie) et toutes les entrées de la terminologie apparaissant dans un document de la discipline sont sélectionnés comme termes-clés candidats de ce document [5]. cette terminologie sont extraites comme termes-clés candidats. Contrairement à la méthode de sélection des plus longues séquences de noms et d'adjectifs, la sélection des candidats termes de TermSuite se fonde sur un travail de spécification linguistique et terminologique des termes. Les patrons grammaticaux utilisés par TermSuite sont donc plus précis (p. ex. `/NOM à NOM/`, `/NOM en NOM/`, `/NOM à NOM ADJ/`, etc.) et de longueur plus restreinte puisque les structures à deux ou trois mots lexicaux sont privilégiées.

Exemples de candidats termes sélectionnés à partir de « bassin moyen du Don » dans la notice d'archéologie de la figure 1 : « bassin », « Don », « bassin moyen » et « bassin moyen du Don ».

3.3 Ordonnancement des termes-clés candidats

Un grand nombre de méthodes sont proposées dans la catégorie des méthodes non-supervisées. Parmi elles, les méthodes d'ordonnancement TF-IDF (Spärck Jones, 1972) et TopicRank (Bougouin *et al.*, 2013). De part sa simplicité et sa robustesse, la méthode TF-IDF s'impose comme la méthode de référence pour l'extraction non-supervisée de termes-clés [6], tandis que les méthodes à base de graphe, telles que TopicRank, suscitent un intérêt grandissant. En effet, les graphes permettent de représenter simplement et efficacement les unités textuelles d'un document et leurs relations en son sein. De plus, ils bénéficient de nombreuses études théoriques donnant lieu à des outils et algorithmes efficaces pour résoudre divers problèmes. TF-IDF et TopicRank ont un fonctionnement très différent, ce qui nous permet par la suite d'identifier les facteurs qui influent sur la difficulté de l'extraction automatique de termes-clés.

La méthode **TF-IDF** consiste à extraire en tant que termes-clés les candidats dont les mots sont importants. Un score d'importance (TF-IDF) est attribué à chaque mot des candidats et l'importance d'un candidat est calculé par la somme du score d'importance de ses mots. Selon TF-IDF, un mot est considéré important dans un document s'il y est fréquent (TF élevé) et s'il a une forte spécificité (IDF élevé). Cette dernière est déterminée à partir d'une collection de documents [7] : moins il y a de documents qui contiennent le mot, plus forte est sa spécificité.

TopicRank (Bougouin *et al.*, 2013) extrait les termes-clés qui représentent les sujets les plus importants d'un document. Tout d'abord, TopicRank groupe les termes-clés candidats selon leur appartenance à un sujet, représente les documents sous la forme d'un graphe de sujets et ordonne les sujets selon leur importance dans le graphe (Mihalcea & Tarau, 2004). Enfin, le terme-clé candidat le plus représentatif d'un sujet, celui qui apparaît en premier dans le document, est extrait en tant que terme-clé [8].

TopicRank groupe les termes-clés candidats selon une mesure de similarité lexicale (cf. équation 1). Cependant, TermSuite fournit un groupement terminologique des termes et de leurs variantes. Lorsque les termes-clés candidats sont ceux extraits avec TermSuite, nous tirons profit de ce groupement terme/variantes à la place de celui fondé sur la similarité lexicale. Tenant compte du groupement (moins naïf) de TermSuite, TopicRank distingue alors les candidats « Kostienki 11/II/ » et « Kostienki 21/III/ » qui représentent des faciès différents (cf. figure 1).

$$\text{similarité}(c_1, c_2) = \frac{\|c_1 \cap c_2\|}{\|c_1 \cup c_2\|}, \tag{1}$$

où c_1 et c_2 sont deux termes-clés candidats représentés par des sacs de mots.

5. L'extraction terminologique effectuée depuis tous les documents de chaque collection permet une meilleure précision lors de la détection des variantes des termes. De plus, la taille des collections entre 80 000 et 150 000 mots est faible pour une extraction terminologique, mais ceci est compensé par le haut degré de densité terminologique des collections.

6. Notons qu'une variante de la pondération TF-IDF est utilisée en Recherche d'Information (Robertson *et al.*, 1998; Claveau, 2012, Okapi). Bien que cette variante est jugée plus efficace en Recherche d'Information, celle-ci n'a, à notre connaissance, jamais été employée pour l'extraction automatique de termes-clés. Notre objectif n'étant pas de trouver la meilleure méthode d'extraction de termes-clés, nous utilisons la méthode originale.

7. Dans ce travail, nous utilisons la collection dont est extrait le document.

8. Si nécessaire, les termes-clés extraits sont pondérés et ordonnés selon le score d'importance de leur sujet respectif

4 Expériences

Dans cette section, nous présentons les expériences menées dans le but d'observer l'échelle de difficulté pour l'extraction automatique de termes-clés en domaines de spécialité à partir des méthodes TF-IDF et TopicRank et en fonction des candidats qui sont sélectionnés : $\{1..3\}$-grammes filtrés, plus longues séquences de noms et d'adjectifs et candidats termes.

4.1 Mesure d'évaluation

Afin de mesurer l'échelle de difficulté pour l'extraction automatique de termes-clés en domaines de spécialité, nous utilisons la MAP (*Mean Average Precision*), qui mesure la capacité d'une méthode à ordonner correctement les termes-clés de référence parmi tous les termes-clés candidats, c'est-à-dire à extraire en premier des candidats qui sont présents dans la liste des termes-clés de référence (cf. équation 2). Alors qu'il est plus courant d'utiliser la précision, le rappel et la f-mesures pour comparer les méthodes entre elles, notre choix se porte sur la MAP à cause du nombre variable de termes-clés de référence assignés aux documents par discipline (de 8,0 en linguistique à 16,6 en archéologie). La MAP étant appliquée à tous les candidats ordonnés et non pas à un sous ensemble (p. ex. les 10 premiers, pour la précision, le rappel et la f-mesure), il ne peut y avoir de biais lorsque nous comparons l'extraction de termes-clés entre deux disciplines.

$$\text{MAP} = \frac{1}{\|\text{DOCUMENTS}\|} \sum_{d \in \text{DOCUMENTS}} \frac{\sum_{t_i \in \text{extraction}(d) \,\cap\, \text{référence}(d)} \text{précision@}i}{\|\text{référence}(d)\|} \tag{2}$$

où :
- extraction(d) founie l'ensemble ordonné des termes-clés candidats t_i de rang i pour le document d,
- référence(d) fournie l'ensemble des termes-clés de référence du document d,
- précision@i représente la précision de l'extraction calculée au rang i,
- DOCUMENTS est l'ensemble des documents de la collection pour laquelle les termes-clés sont extraits.

En accord avec l'évaluation menée dans les travaux précédents, nous considérons correcte l'extraction d'une variante flexionnelle d'un terme-clé de référence (Kim *et al.*, 2010). Les opérations de comparaison entre les termes-clés de référence et les termes-clés extraits sont donc effectuées à partir de la racine des mots qui les composent en utilisant la méthode de racinisation de Porter (1980).

4.2 Résultats

La figure 3 montre la performance des méthodes d'extraction automatique de termes-clés lorsque les candidats sélectionnés sont soit les $\{1..3\}$-grammes filtrés, soit les plus longues séquences de noms et d'adjectifs, soit tous les candidats termes extraits par TermSuite (sans filtrage). Notre hypothèse de départ selon laquelle la tâche d'extraction de termes-clés présente un degré de difficulté différent selon la discipline scientifique se vérifie. L'archéologie est la discipline pour laquelle la tâche d'extraction automatique de termes-clés est la moins difficile, la chimie étant la discipline la plus difficile, précédée par la psychologie, les sciences de l'information et la linguistique. Quelle que soit la discipline traitée, nous pouvons aussi observer la faible performance des méthodes d'extraction de termes-clés (cf. exemple figure 4). Ceci peut s'expliquer par le faible rappel maximum pouvant être atteint, ainsi que par l'évaluation stricte qui n'accepte pas les correspondances partielles (p. ex. « articles » et « articles de recherche » qui dans le contexte de la notice de la figure 4 représentent le même concept).

Globalement, les meilleurs résultats sont obtenus avec la méthode TF-IDF. De plus, bien que dans le meilleur cas elle soit compétitive avec TF-IDF, la méthode TopicRank n'est pas stable. Lorsque les $\{1..3\}$-grammes sont utilisés comme candidats nous observons une forte dégradation des résultats de TopicRank, alors que la dégradation des résultats de TF-IDF est plus modérée. Cette différence de comportement face à un ensemble de termes-clés candidats de mauvaise qualité s'explique par le fait que le groupement en sujets de TopicRank n'est pas adapté pour de tels candidats et aussi parce que TF-IDF tire profit de la spécificité des mots (IDF), lui permettant, contrairement à TopicRank, de ne pas attribuer un fort poids aux candidats erronés tels que « d' » (cf. figure 4). En ce qui concerne les résultats obtenus avec les deux autres méthodes de sélection des termes-clés candidats, les performances sont meilleures avec les plus longues séquences de noms et d'adjectifs. La différence de performance observée avec ces deux méthodes de sélection est principalement

liée à la richesse des patrons grammaticaux utilisés par TermSuite. En effet, ses patrons grammaticaux contenant des déterminants et des prépositions ne reflètent qu'une infime quantité de termes-clés de référence (3,5 %) et ont donc pour effet d'ajouter plus de bruit que de candidats positifs.

5 Discussion

À partir des expériences de la section 4, nous constatons la même échelle de difficulté quelque soit la méthode employée (cf. figure 5), ce qui montre que notre hypothèse de départ est valide. Il est toutefois important de noter qu'en observant les statistiques présentées dans le tableau 1, nous pouvons déduire la même échelle de difficulté à partir du rappel maximum. Cependant, le rappel maximum ne peut être obtenu en dehors d'un contexte expérimental. Dans cette section, nous nous fondons sur la nature des collections de données et sur les résultats de l'extraction non-supervisée de termes-clés pour déterminer quels sont les facteurs qui influent sur la difficulté de cette tâche.

Dans un premier temps, nous constatons que la pondération fondée sur la spécificité des mots améliore la stabilité (la robustesse) des méthodes d'extraction de termes-clés qui l'utilisent. Nous en déduisons que la nature linguistique des termes-clés utilisés dans une discipline est un facteur qui influe sur la difficulté de l'extraction des termes-clés. Ainsi, une forte tendance à l'usage de composés syntagmatiques constitués de mots centraux dans la discipline, tels que « réaction » en Chimie (p. ex. « réaction topotactique » et « réaction sonochimique ») ou encore le mot « social », qui est fréquemment utilisé en psychologie (p. ex. « interaction sociale » et « environnement social »), augmente la difficulté de l'extraction des termes-clés.

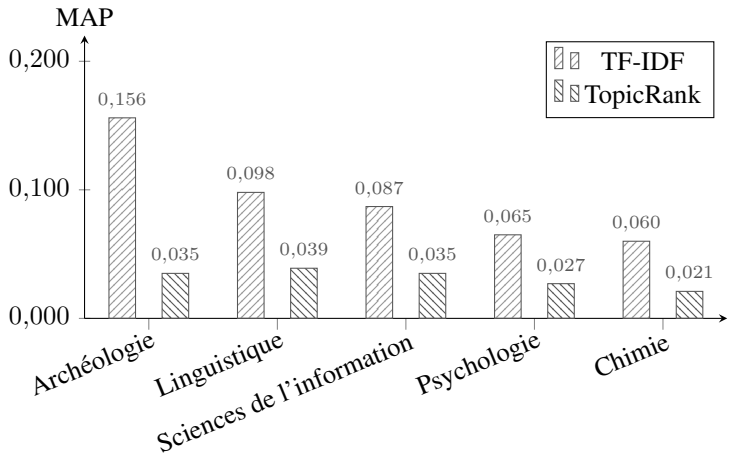
Dans un second temps, nous observons, sauf dans le cas de la psychologie, qu'il y a une correspondance entre l'ordre des disciplines selon la taille des résumés des notices et leur ordre dans l'échelle de difficulté. Ceci s'explique par la façon dont est organisé le discours dans les notices. Si nous prenons les notices d'archéologie, par exemple, celles-ci sont très détaillées. Il est par conséquent aisé d'établir des relations entre les concepts afin de déterminer quels sont ceux les plus importants, à la manière de TopicRank. À l'inverse, les notices de chimie, représentant principalement des comptes rendus d'expériences s'adressent à un lecteur expert pour lequel il est uniquement nécessaire de décrire le contexte expérimental. L'absence de détails dans les notices de certaines disciplines est donc un facteur qui influe sur la difficulté de l'extraction automatique de termes-clés, difficulté qui peut a priori être détectée à partir de la taille des notices.

6 Conclusion et perspectives

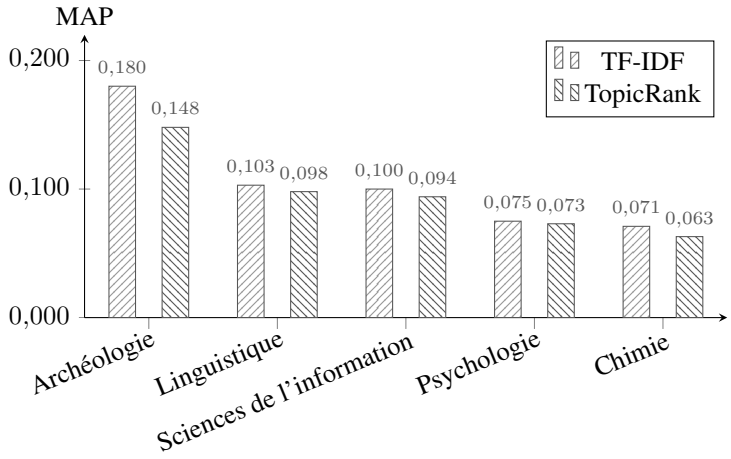
Dans cet article, nous nous intéressons à la tâche d'extraction automatique de termes-clés dans les documents scientifiques et émettons l'hypothèse que sa difficulté est variable selon la discipline des documents traités. Pour vérifier cette hypothèse, nous disposons de notices bibliographiques réparties dans cinq disciplines (archéologie, linguistique, sciences de l'information, psychologie et chimie) auxquelles nous appliquons six systèmes d'extraction automatique de termes-clés différents. En comparant les termes-clés extraits par chaque système avec les termes-clés de référence assignés aux notices dans des conditions réels d'indexation, notre hypothèse se vérifie et nous observons l'échelle suivante (de la discipline la plus facile à la plus difficile) : 1. Archéologie ; 2. Linguistique ; 3. Sciences de l'information ; 4. Psychologie ; 5. Chimie.

À l'issue de nos expériences et de nos observations du contenu des notices, nous constatons deux facteurs ayant un impact sur la difficulté de la tâche d'extraction automatique de termes-clés. Tout d'abord, nous observons que l'organisation du résumé peut aider l'extraction de termes-clés. Un résumé riche en explications et en mises en relations des différents concepts est moins difficile à traiter qu'un résumé énumératif pauvre en explications. Ensuite, le vocabulaire utilisé dans une discipline peut influer sur la difficulté à extraire les termes-clés des documents de cette discipline. Si le vocabulaire spécifique contient des composés syntagmatiques dont certains éléments sont courants dans la discipline, alors il peut être plus difficile d'extraire les termes-clés des documents de cette discipline.

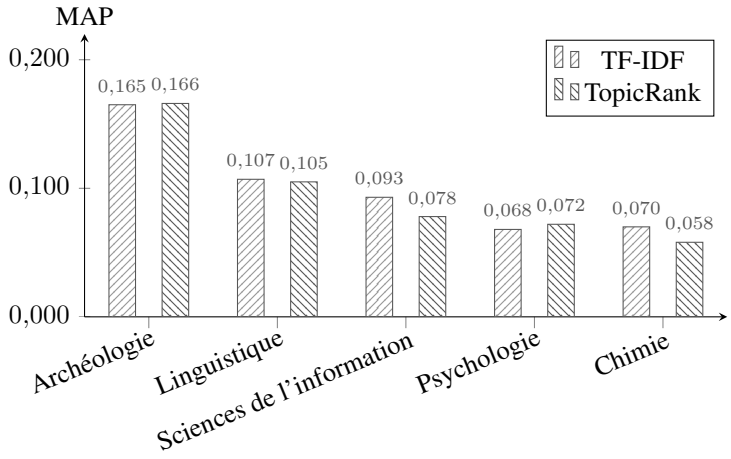
Des deux facteurs identifiés émergent plusieurs perspectives de travaux futurs. Il peut être intéressant d'analyser le discours des documents afin de mesurer, en amont, le degré de difficulté de l'extraction de termes-clés. Avec une telle connaissance, nous pourrions proposer une méthode capable de s'adapter au degré de difficulté en ajustant automatiquement son paramètrage. Cependant, l'analyse que nous proposons dans cet article se fonde uniquement sur le contenu de notices appartenant à cinq disciplines. Il serait pertinent d'étendre cette analyse au contenu intégral des documents scientifiques, ainsi que d'élargir le panel de disciplines utilisées dans ce travail, afin d'établir des catégories de disciplines plus ou moins difficiles à traiter (p. ex. la chimie fait partie des disciplines expérimentales, qui sont difficiles à traiter). Nous



(a) {1..3}-grammes



(b) `/(NOM | ADJ)+/`



(c) Candidats termes

FIGURE 3 – Performances des méthodes d'extraction de termes-clés en domaines de spécialité à partir de différents type de candidats.

Termes techniques et marqueurs d'argumentation : pour débusquer l'argumentation cachée dans les articles de recherche *Linguistique*

Les articles de recherche présentent les résultats d'une expérience qui modifie l'état de la connaissance dans le domaine concerné. Le lecteur néophyte a tendance à considérer qu'il s'agit d'une simple description et à passer à côté de l'argumentation au cours de laquelle le scientifique cherche à convaincre ses pairs de l'innovation et de l'originalité présentées dans l'article et du bien-fondé de sa démarche tout en respectant la tradition scientifique dans laquelle il s'insère. Ces propriétés spécifiques du discours scientifique peuvent s'avérer un obstacle supplémentaire à la compréhension, surtout lorsqu'il s'agit d'un article en langue étrangère. C'est pourquoi il peut être utile d'incorporer dans l'enseignement des langues de spécialité une sensibilisation aux marqueurs linguistiques (terminologiques et argumentatifs), qui permettent de dépister le développement de cette rhétorique. Les auteurs s'appuient sur deux articles dans le domaine de la microbiologie.

Termes-clés de référence : Langue scientifique*, argumentation*, rhétorique*, langue de spécialité*, enseignement des langues*, linguistique appliquée*, discours scientifique*, article de recherche.

Termes-clés extraits :

{1..3}-grammes

TF-IDF : Argumentation, scientifique, articles, d' argumentation, l' argumentation, tradition scientifique, **discours scientifique**, marqueurs.

TopicRank : Articles, d', qu' il s', débusquer l' argumentation, **articles de recherche**, scientifique, s' agit d', marqueurs d' argumentation.

`/(NOM | ADJ)+/`

TF-IDF : Argumentation, scientifique, articles, tradition scientifique, **discours scientifique**, marqueurs, microbiologie, domaine.

TopicRank : Article, **argumentation**, recherche, marqueurs, domaine, langue étrangère, scientifique, résultats.

Candidats termes

TF-IDF : Argumentation, scientifique, tradition scientifique, **discours scientifique**, marqueurs, microbiologie, néophyte, marqueurs d' argumentation.

TopicRank : Argumentation, marqueurs, **articles de recherche**, scientifique, techniques, termes, article, langue.

FIGURE 4 – Exemple d'extraction automatique de (huit) termes-clés à partir de la notice de linguistique présentée dans la figure 1. Les termes-clés de référence soulignés sont ceux qui occurrent dans le titre ou le résumé de la notice. Les termes-clés de référence marqués d'une * font partie des termes-clés contrôlés. Les termes-clés extraits mis en gras sont les termes-clés correctement extraits.

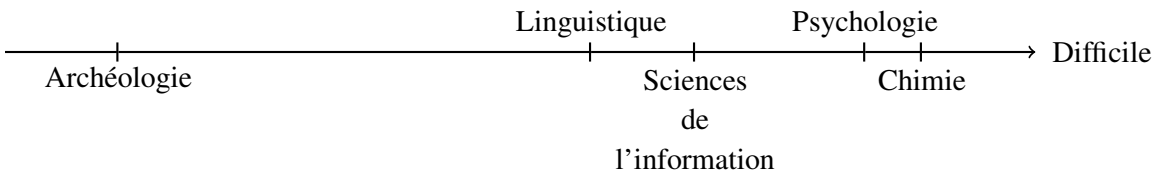


FIGURE 5 – Échelle de difficulté disciplinaire, de la discipline la moins difficile à la discipline la plus difficile à traiter par les méthodes d'extraction automatique de termes-clés.

observons aussi que le vocabulaire utilisé dans une discipline, en particulier celui utilisé pour les termes-clés, peut rendre la tâche d'extraction automatique de termes-clés plus difficile. Il est donc important de bénéficier de resources telles que des thésaurus pour permettre à une méthode d'extraction de termes-clés de s'adapter au domaine. Pour TopicRank, par exemple, avoir connaissance de la terminologie utilisée dans une discipline peut améliorer le choix du terme-clé le plus représentatif d'un sujet. Enfin, il serait intéressant de penser la tâche d'extraction de termes-clés comme une tâche d'extraction d'information pour le remplissage d'un formulaire. En archéologie, par exemple, il pourrait s'agir d'extraire les informations géographiques (pays, régions, etc.), chronologiques (période, culture, etc.), ou encore environnementales (animaux, végétaux, etc.).

Remerciements

Ce travail a bénéficié d'une aide de l'Agence Nationale de la Recherche portant la référence (ANR-12-CORD-0029).

Références

BIRD S., KLEIN E. & LOPER E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

BOUGOUIN A., BOUDIN F. & DAILLE B. (2013). Topicrank : Graph-Based Topic Ranking for Keyphrase Extraction. In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP)*, p. 543–551, Nagoya, Japan : Asian Federation of Natural Language Processing.

CLAVEAU V. (2012). Vectorisation, Okapi et calcul de similarité pour le TAL : pour oublier enfin le TF-IDF (Vectorization, Okapi and Computing Similarity for NLP : Say Goodbye to TF-IDF) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, Volume 2 : TALN*, p. 85–98, Grenoble, France : ATALA/AFCP.

DENIS P. & SAGOT B. (2009). Coupling an Annotated Corpus and a Morphosyntactic Lexicon for State-of-the-Art POS Tagging with Less Human Effort. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC)*, p. 110–119, Hong Kong : City University of Hong Kong.

D'AVANZO E. & MAGNINI B. (2005). A Keyphrase-Based Approach to Summarization : the LAKE System at DUC-2005. In *Proceedings of DUC 2005 Document Understanding Conference*.

HAN J., KIM T. & CHOI J. (2007). Web Document Clustering by Using Automatic Keyphrase Extraction. In *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, p. 56–59, Washington, DC, USA : IEEE Computer Society.

HASAN K. S. & NG V. (2010). Conundrums in Unsupervised Keyphrase Extraction : Making Sense of the State-of-the-Art. In *Proceedings of the 23rd International Conference on Computational Linguistics : Posters*, p. 365–373, Stroudsburg, PA, USA : Association for Computational Linguistics.

HASAN K. S. & NG V. (2014). Automatic Keyphrase Extraction : A Survey of the State of the Art. In *Proceedings of the Association for Computational Linguistics (ACL)*, Baltimore, Maryland : Association for Computational Linguistics.

HULTH A. (2003). Improved Automatic Keyword Extraction Given More Linguistic Knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, p. 216–223, Stroudsburg, PA, USA : Association for Computational Linguistics.

KIM S. N., MEDELYAN O., KAN M.-Y. & BALDWIN T. (2010). SemEval-2010 task 5 : Automatic Keyphrase Extraction from Scientific Articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, p. 21–26, Stroudsburg, PA, USA : Association for Computational Linguistics.

MEDELYAN O. & WITTEN I. H. (2008). Domain-Independent Automatic Keyphrase Indexing with Small Training Sets. *Journal of the American Society for Information Science and Technology*, **59**(7), 1026–1040.

MIHALCEA R. & TARAU P. (2004). TextRank : Bringing Order Into Texts. In DEKANG LIN & DEKAI WU, Eds., *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, p. 404–411, Barcelona, Spain : Association for Computational Linguistics.

PAROUBEK P., ZWEIGENBAUM P., FOREST D. & GROUIN C. (2012). Indexation libre et contrôlée d'articles scientifiques. Présentation et résultats du défi fouille de textes DEFT2012 (Controlled and Free Indexing of Scientific Papers. Presentation and Results of the DEFT2012 Text-Mining Challenge) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, Workshop DEFT 2012 : DÉfi Fouille de Textes (DEFT 2012 Workshop : Text Mining Challenge)*, p. 1–13, Grenoble, France : ATALA/AFCP.

PORTER M. F. (1980). An Algorithm for Suffix Stripping. *Program : Electronic Library and Information Systems*, **14**(3), 130–137.

ROBERTSON S. E., WALKER STEVE & HANCOCK-BEAULIEU MICHELINE (1998). Okapi at TREC-7 : Automatic Ad Hoc, Filtering, VLC and Interactive Track. In *Proceedings of the Text REtrieval Conference (TREC)*, p. 199–210.

ROCHETEAU J. & DAILLE B. (2011). TTC TermSuite - A UIMA Application for Multilingual Terminology Extraction from Comparable Corpora. In *Proceedings of the IJCNLP 2011 System Demonstrations*, p. 9–12, Chiang Mai, Thailand : Asian Federation of Natural Language Processing.

SPÄRCK JONES K. (1972). A Statistical Interpretation of Term Specificity and its Application in Retrieval. *Journal of Documentation*, **28**(1), 11–21.

WITTEN I. H., PAYNTER G. W., FRANK E., GUTWIN C. & NEVILL MANNING C. G. (1999). KEA : Practical Automatic Keyphrase Extraction. In *Proceedings of the 4th ACM Conference on Digital Libraries*, p. 254–255, New York, NY, USA : ACM.

Étiquetage en rôles événementiels fondé sur l'utilisation d'un modèle neuronal

Emanuela Boroş[1,2] Romaric Besançon[1] Olivier Ferret[1] Brigitte Grau[2, 3]
(1) CEA, LIST, Laboratoire Vision et Ingénierie des Contenus, F-91191, Gif-sur-Yvette
(2) LIMSI, Rue John von Neumann, Campus Universitaire d'Orsay, F-91405 Orsay cedex
(3) ENSIIE, 1 square de la résistance F-91025 Évry cedex
{emanuela.boros,romaric.besancon,olivier.ferret}@cea.fr, brigitte.grau@limsi.fr

Résumé. Les systèmes d'extraction d'information doivent faire face depuis toujours à une double difficulté : d'une part, ils souffrent d'une dépendance forte vis-à-vis du domaine pour lesquels ils ont été développés ; d'autre part, leur coût de développement pour un domaine donné est important. Le travail que nous présentons dans cet article se focalise sur la seconde problématique en proposant néanmoins une solution en relation avec la première. Plus précisément, il aborde la tâche d'étiquetage en rôles événementiels dans le cadre du remplissage de formulaire (*template filling*) en proposant pour ce faire de s'appuyer sur un modèle de représentation distribuée de type neuronal. Ce modèle est appris à partir d'un corpus représentatif du domaine considéré sans nécessiter en amont l'utilisation de prétraitements linguistiques élaborés. Il fournit un espace de représentation permettant à un classifieur supervisé traditionnel de se dispenser de l'utilisation de traits complexes et variés (traits morphosyntaxiques, syntaxiques ou sémantiques). Par une série d'expérimentations menées sur le corpus de la campagne d'évaluation MUC-4, nous montrons en particulier que cette approche permet de dépasser les performances de l'état de l'art et que cette différence est d'autant plus importante que la taille du corpus d'entraînement est faible. Nous montrons également l'intérêt de l'adaptation de ce type de modèle au domaine traité par rapport à l'utilisation de représentations distribuées à usage générique.

Abstract. Information Extraction systems must cope with two problems : they heavily depend on the considered domain but the cost of development for a domain-specific system is important. We propose a new solution for role labeling in the event-extraction task that relies on using unsupervised word representations (*word embeddings*) as word features. We automatically learn domain-relevant distributed representations from a domain-specific unlabeled corpus without complex linguistic processing and use these features in a supervised classifier. Our experimental results on the MUC-4 corpus show that this system outperforms state-of-the-art systems on this event extraction task, especially when the amount of annotated data is small.We also show that using word representations induced on a domain-relevant dataset achieves better results than using more general word embeddings.

Mots-clés : Extraction d'information, extraction de rôles événementiels, modèles de langage neuronaux.

Keywords: Information extraction, event role filler detection, neural language models.

1 Introduction

Un enjeu majeur de l'Extraction d'Information (EI) consiste à aider un utilisateur à identifier rapidement des événements ainsi que leurs entités descriptives dans de très grands volumes de documents. L'extraction d'événements peut porter sur des domaines variés. Dans le domaine médical et biologique, la notion d'événement est utilisée pour désigner par exemple le changement d'état d'une molécule en biologie, ou encore l'ensemble des informations concernant l'administration d'un traitement en médecine (Cohen *et al.*, 2009; Yakushiji *et al.*, 2001; Chun *et al.*, 2005). Dans le domaine de l'économie et la finance, les centres d'intérêt concernent les fusions, acquisitions et échanges d'entreprises ou de produits (Hung *et al.*, 2010; Michaely *et al.*, 1995).

Un événement est décrit par un ensemble de participants (*i.e.* des attributs ou rôles) dont les valeurs sont des extraits de texte, correspondant à des entités nommées ou des entités du domaine. Par exemple, un acte terroriste est un événement dont les participants sont les auteurs, les victimes ou encore les cibles. En domaine biomédical, un type d'événement largement étudié est celui des interactions où les rôles désignent des protéines ou des gènes, des médicaments ou autres

molécules. Cette problématique est issue des campagnes d'évaluation MUC [1] (Grishman & Sundheim, 1996), TREC (Voorhees & Tong, 2011) et ACE (Strassel *et al.*, 2008) qui ont fortement contribué à l'évolution du domaine. Dans cet article, nous nous intéressons plus spécifiquement à la reconnaissance des entités et leur étiquetage en rôle. Cette tâche est complexe et recouvre des problématiques telles que la reconnaissance d'entités nommées, la reconnaissance de rôles sémantiques ou l'extraction de relations binaires.

Pour cette tâche, beaucoup de systèmes proposent des méthodes d'extraction de patrons ou de génération de règles fondées sur le contexte environnant, local et global (Patwardhan & Riloff, 2009; Huang & Riloff, 2011). Les méthodes d'acquisition de ces patrons incluent des approches par amorçage (Huang & Riloff, 2012a; Yangarber *et al.*, 2000), de l'apprentissage faiblement supervisé (Huang & Riloff, 2011; Sudo *et al.*, 2003; Surdeanu *et al.*, 2006), de l'apprentissage supervisé (Chieu *et al.*, 2003; Freitag, 1998; Bunescu & Mooney, 2004; Patwardhan & Riloff, 2009), et autres variations. Les patrons appris par ces approches sont ensuite utilisés pour reconnaître et étiqueter, dans de nouveaux documents, des extraits de textes en tant que valeurs d'attributs.

Toutes ces méthodes reposent sur une part assez importante d'annotations manuelles couplée à l'utilisation intensive de connaissances linguistiques et les performances obtenues sont donc en rapport avec la possibilité de mettre en œuvre cette masse de connaissances ainsi que la capacité à définir les ensembles de traits en entrée des classifieurs. De plus, la bonne application de ces méthodes nécessite de connaître a priori le domaine d'application. Il devient ainsi difficile d'appliquer efficacement une méthode donnée sur un domaine différent.

Dans ce travail, nous abordons la tâche d'étiquetage d'entités en rôles décrivant un événement, que nous nommons étiquetage en rôles événementiels, par l'apprentissage automatique de traits pertinents qui ne nécessite qu'un nombre limité de connaissances préalables. A cette fin, des représentations de mots (*word embeddings*) sont induites par application non supervisée d'un réseau de neurone comme dans (Bengio *et al.*, 2006; Collobert & Weston, 2008) sur des données brutes. Les valeurs d'attributs relatives aux exemples connus d'événements à extraire sont annotées dans des textes et transformées à partir des représentations apprises pour entraîner un classifieur permettant de prédire l'étiquette du rôle rempli. Notre objectif est double : (1) montrer que des représentations de mots apprises de façon non supervisée ont une capacité de généralisation et de représentation du sens qui les rend compétitives sur la tâche d'étiquetage en rôles événementiels, (2) montrer que ces représentations sont évolutives et robustes lorsqu'on fait varier la taille des données d'apprentissage.

Nous avons évalué notre approche sur les données issues de MUC-4 (Lehnert *et al.*, 1992) qui portent sur des actes terroristes, et donc sur la reconnaissances des auteurs, victimes et cibles, et nous obtenons des résultats supérieurs à ceux des méthodes état de l'art sur ces mêmes données (Huang & Riloff, 2011, 2012a; Patwardhan & Riloff, 2009).

Après avoir présenté l'état de l'art en extraction d'événements et en apprentissage de représentations dans la section 2, la suite de cet article décrit plus précisément notre approche en section 3. Les données d'expérimentation et les résultats de l'évaluation sont ensuite présentés dans la section 4.

2 État de l'art

Bien qu'il n'existe pas une manière unique d'aborder l'extraction d'événements à partir de textes, celle-ci est reconnue comme un problème complexe que l'on décompose en différentes tâches prototypiques [2] : détection des mentions d'événement ; extraction des candidats au remplissage des rôles ; rattachement local, souvent au niveau phrastique, des candidats aux mentions d'événement ; fusion au niveau textuel des candidats au remplissage des rôles. Le problème que nous abordons ici est celui de la détection des candidats au remplissage des rôles d'un événement, tâche que l'on peut considérer également comme une annotation de phrases en rôles événementiels. Les candidats sont en toute généralité des groupes nominaux, dont certains peuvent correspondre à des entités nommées.

Deux grands types d'approches ont été proposés pour la tâche d'extraction d'événements : les approches fondées sur l'application de patrons (Krupka *et al.*, 1991; Hobbs *et al.*, 1992; Riloff, 1996a,b; Yangarber *et al.*, 2000)) et les approches par apprentissage (Chieu *et al.*, 2003; Freitag, 1998; Huang & Riloff, 2011; Patwardhan & Riloff, 2009; Yangarber *et al.*, 2000; Surdeanu *et al.*, 2006)).

Les patrons sont acquis à partir de textes par application de règles reposant sur des connaissances syntaxiques, extraites d'un arbre syntaxique par exemple, et sémantiques pour identifier les rôles. Les premiers systèmes (Krupka *et al.*, 1991;

1. MUC 1-7 Message Understanding Conferences de 1987 à 1998 organisées par le DARPA.
2. La décomposition que nous faisons ici est essentiellement fonctionnelle et ne fait pas apparaître les liens de dépendance pouvant exister entre ces différentes tâches.

Hobbs *et al.*, 1992; Riloff, 1996a) sont issus des conférences MUC. AutoSlog-TS (Riloff, 1996a), utilisé comme système de base dans nos évaluations et qui est une version améliorée de AutoSlog (Riloff, 1996b), propose une séparation en textes pertinents et non pertinents et un ordonnancement des patrons extraits. Le principal inconvénient de ces systèmes est qu'ils font appel à une vérification manuelle pour sélectionner les patrons, qui peut s'avérer coûteuse.

L'intérêt porté aux approches par apprentissage s'est largement développé et a donné lieu à de nouveaux systèmes, qui reposent sur des méthodes d'apprentissage complètement supervisées (Chieu *et al.*, 2003; Freitag, 1998; Bunescu & Mooney, 2004; Patwardhan & Riloff, 2009) ou faiblement supervisées (Huang & Riloff, 2011; Sudo *et al.*, 2003; Surdeanu *et al.*, 2006). La dépendance des systèmes à l'existence d'annotations riches des textes s'est relâchée avec l'apparition des techniques d'amorçage (Huang & Riloff, 2012a; Yangarber *et al.*, 2000).

Par exemple, le système ALICE (Chieu *et al.*, 2003) est fondé sur différents algorithmes d'apprentissage, utilisant un riche ensemble de traits syntaxiques et sémantiques, et montre que ces types de traits permettent d'améliorer les performances des systèmes. Les auteurs de (Bunescu & Mooney, 2004) proposent une variante des CRFs (Conditional Random Fields) pour exploiter les relations entre traits et en montrent l'intérêt sur la tâche d'extraction d'événements biomédicaux.

Différentes approches ont exploré l'importance du contexte environnant pour reconnaître des valeurs de rôles. GLACIER (Patwardhan & Riloff, 2009) va au-delà de l'analyse du contexte local de la mention d'un événement (*i.e.* la proposition) pour analyser un second contexte plus large au niveau de la phrase. En reprenant cette idée, nous explorons aussi un contexte assez large pour rechercher des valeurs de rôles lors de l'apprentissage du modèle de langue. TIER (Huang & Riloff, 2011) a exploré le fait d'écarter un contexte s'il est non pertinent, même si dans certaines situations une phrase non pertinente peut mentionner les suites d'un événement où certains rôles sont précisés. TIER repose sur une suite de traitements héritée des systèmes précédents partant de la reconnaissance de contextes spécifiques à des rôles par différentes couches de classifieurs et finissant, au niveau le plus bas, par l'extraction des valeurs des rôles. L'amélioration de TIER par l'usage de co-références et de relations de discours est étudiée dans (Huang & Riloff, 2012b,a).

De tous ces travaux, on peut voir que les recherches en extraction d'événements vont vers l'ajout de traits riches donnés en entrée de chaînes de classifieurs, et vers une exploration du contexte des entités à étiqueter. Le but de (Huang & Riloff, 2012b) est de pouvoir reconnaître des transitions et des relations de discours dans le texte de manière à pouvoir mieux identifier les contextes d'apparition des rôles pour un événement donné. Les candidats aux rôles liés à un événement sont identifiés indépendamment, par une approche montante, puis les contextes sélectionnés en mettant en oeuvre des connaissances pour déterminer la cohésion textuelle. PIPER (Patwardhan & Riloff, 2007; Patwardhan, 2010) est construit à partir d'une classification des phrases qui distingue les régions pertinentes et non pertinentes et apprend des patrons d'extraction pertinents pour le domaine selon une mesure d'affinité sémantique. On peut aussi ajouter que, même si l'amélioration des performances est réelle, ces ajouts peuvent rendre les systèmes très lents et non utilisables dans des applications à grande échelle. C'est pourquoi TIER$_{light}$ (Huang & Riloff, 2012a) propose de diminuer le recours à des annotations lourdes, pour faciliter le passage d'un domaine à un autre, par l'application de techniques d'amorçage pour l'étiquetage en rôles.

Notre approche partage avec ces approches l'importance donnée au contexte des mots impliqués dans des valeurs de rôles. Cependant, alors que ces systèmes reposent sur la conception de riches ensembles de traits à donner en entrée des classifieurs, notre approche réduit cette complexité en donnant seulement les mots bruts en entrée d'un réseau de neurones. Les traits sont appris automatiquement et sont réutilisés dans la tâche d'étiquetage en rôles événementiels ; ils permettent de plus d'obtenir de meilleurs résultats à partir de ces seules données. Ce type d'approche, proposé dans des travaux comme (Bengio *et al.*, 2006; Collobert & Weston, 2008; Turian *et al.*, 2010), a montré des résultats intéressants sur de nombreuses tâches en traitement automatique des langues mais n'a jamais été appliqué à l'extraction d'événements.

3 Méthode

3.1 Principes

À l'instar de (Huang & Riloff, 2012b), la tâche d'étiquetage en rôles événementiels est réalisée comme une tâche indépendante des autres tâches mentionnées section 2. Son objectif est de produire un ensemble assez large de candidats qui seront ensuite filtrés par les contraintes de rattachement aux événements, soit au niveau local, soit au niveau global. À la différence de (Jean-Louis *et al.*, 2011), nous ne faisons pas l'hypothèse que ces candidats se limitent à des entités nommées et nous nous ne faisons pas non plus l'hypothèse d'une bijection entre le rôle d'un événement et un type d'entité nommée.

Sur le plan méthodologique, nous traitons cette tâche d'étiquetage en rôles événementiels sous l'angle de la classification supervisée. Nous nous appuyons pour ce faire sur la sortie d'un outil générique de découpage des phrases en chunks syntaxiques et nous appliquons un classifieur multiclasse à chaque chunk nominal extrait pour déterminer à quel rôle du type d'événement considéré il est susceptible de se rattacher. Nous avons ainsi une classe par rôle à laquelle s'ajoute une classe correspondant à l'absence de rattachement. L'originalité de l'approche que nous proposons réside dans le type de représentation des chunks nominaux exploité par notre classifieur. Pour ce type de tâche, il est habituel de représenter chaque candidat à un rôle par un ensemble de traits caractérisant différents types d'informations allant des simples mots le constituant jusqu'à son rôle sémantique dans la phrase en passant par la catégorie morphosyntaxique de ses constituants ou son rôle syntaxique. Comme nous l'avons mentionné dans la section précédente, cette approche a un triple inconvénient : elle nécessite un ensemble d'outils élaborés qui ne sont pas toujours disponibles pour une langue donnée ; ces outils n'étant pas parfaits, les informations qu'ils délivrent sont entachées d'un certain taux d'erreur, qui a tendance à être d'autant plus conséquent que l'outil est plus élaboré ; enfin, ces outils sont génériques et donc, la plupart du temps, non adaptés au domaine considéré.

Pour faire face à ces problèmes, nous proposons d'adopter une approche différente, inspirée de travaux tels que (Collobert & Weston, 2008), consistant à projeter les candidats à un rôle événementiel, à partir de leurs mots, dans un espace de représentation défini spécifiquement pour le domaine considéré. Plus précisément, cet espace est construit grâce à un réseau de neurones en reprenant des techniques développées pour l'apprentissage de modèles de langage (Bengio *et al.*, 2003). Outre leur adaptation au domaine, les représentations ainsi élaborées ont l'avantage de pouvoir être comparées et leur proximité dans cet espace est à mettre en relation avec la proximité de leur rôle vis-à-vis du domaine. Une fois construites, ces représentations sont utilisées comme traits dans un classifieur supervisé réalisant l'étiquetage en rôles événementiels, à l'instar des traits habituellement utilisés pour cette tâche.

Nous commençons par détailler la façon dont ces représentations sont construites sur un plan générique avant de préciser leur utilisation et les stratégies mises en œuvre pour les adapter à notre contexte de travail.

3.2 Construction des représentations lexicales distribuées

Notre construction de représentations lexicales distribuées (*word embeddings*) s'appuie sur les principes définis dans (Collobert & Weston, 2008). Ces principes sont eux-mêmes issus de la problématique des modèles de langage neuronaux (Bengio *et al.*, 2003). Dans ce contexte, un réseau de neurones est entraîné à prédire la probabilité d'un mot en fonction de son contexte. L'entraînement d'un tel modèle passe par le traitement d'un large ensemble d'exemples de séquences de mots et une optimisation des paramètres du réseau du point de vue de sa capacité à fournir les meilleures prédictions pour les séquences exemples.

Une des spécificités des réseaux utilisés réside dans la représentation des séquences de mots qu'ils prennent comme entrée et plus spécifiquement des mots composant ces séquences. Dans ce schéma de représentation en effet, un mot n'est plus considéré comme un simple symbole mais possède une représentation distribuée. Cette représentation prend la forme d'un ensemble fixe de dimensions valuées, c'est-à-dire un vecteur de nombres réels de même taille pour tous les mots du vocabulaire considéré. De ce point de vue, cette représentation est proche de représentations issues de méthodes de réduction de dimensions telles que celles produites par l'Analyse Sémantique Latente par exemple (Landauer *et al.*, 1998). À la différence de ces méthodes de réduction de dimensions, qui appliquent une transformation mathématique donnée, les représentations produites dans le cadre des modèles de langage neuronaux sont apprises en relation avec les exemples exploités. Elles sont donc intrinsèquement adaptées à ces derniers. (Collobert & Weston, 2008) a repris ce schéma mais avec la perspective plus générale de construire des représentations lexicales distribuées utilisables pour des tâches autres que la prédiction de la probabilité d'une séquence de mots.

Nous nous inscrivons dans le prolongement de (Collobert & Weston, 2008) pour construire des représentations dédiées à l'étiquetage en rôles événementiels. Cette construction prend la forme de l'apprentissage d'un modèle permettant de différencier de façon générique une séquence de mots issue d'un corpus représentatif d'un domaine et une séquence de mots proche mais ne figurant pas dans le corpus. En pratique, les secondes sont construites par l'altération des premières en changeant un de leurs mots, en l'occurrence celui du milieu. Nous verrons à la section suivante comment s'effectue ce changement. La tâche peut donc être vue comme un test de compatibilité du mot central d'une séquence avec son contexte environnant du point de vue du corpus considéré et donc, de son domaine associé.

Le modèle à apprendre prend plus spécifiquement la forme d'un réseau de neurones à trois couches, comme l'illustre la figure 1, avec une première couche (de gauche à droite) permettant de représenter les séquences en entrée et une couche

finale ayant pour rôle de leur attribuer un score. Les séquences correspondent au contenu d'une fenêtre glissante déplacée sur les textes et composée de $m = 2n+1$ mots. Chaque unité de la couche d'entrée du réseau de la figure 1 ne correspond pas directement à un mot de cette fenêtre mais à une des k dimensions de sa représentation distribuée. La couche d'entrée du réseau est ainsi formée de la concaténation des représentations distribuées des m mots de la fenêtre et contient donc $k \cdot m$ unités.

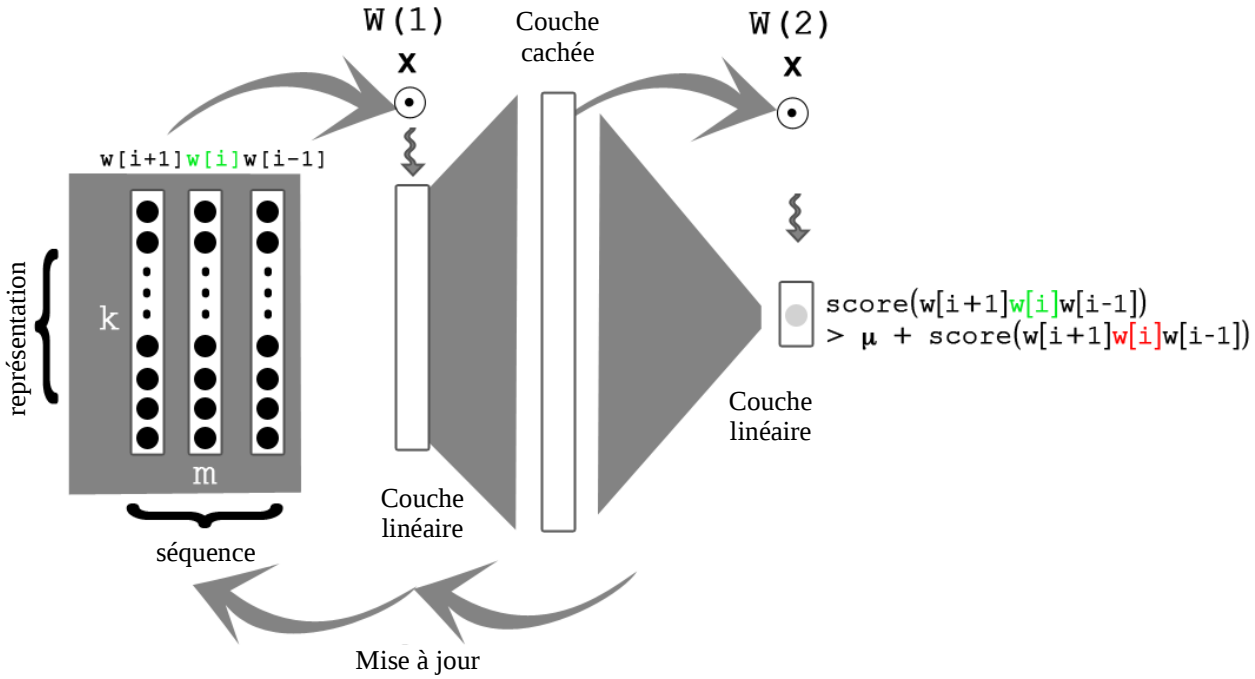


FIGURE 1 – Architecture du réseau de neurones utilisé

Lors de la première phase du processus d'apprentissage, les différentes dimensions de la représentation d'un mot sont initialisées de manière aléatoire selon une loi uniforme. L'activation correspondante est propagée dans le réseau, d'abord vers la couche cachée, puis vers la couche de sortie pour aboutir au calcul d'un score associé à la séquence d'entrée. D'un point de vue plus formel, pour la séquence d'entrée $\langle w_i \rangle = \langle w_{i-n)} \dots, w_{i-1}, w_i, w_{i+1} \dots w_{i+n} \rangle$, on a ainsi :

$$\begin{aligned} score(\langle w_i \rangle) &= b^{(2)} + W^{(2)} h(\langle w_i \rangle) \\ h(\langle w_i \rangle) &= \Phi(b^{(1)} + W^{(1)} \langle w_i \rangle) \end{aligned} \tag{1}$$

$b^{(1)}$, $b^{(2)}$ étant les termes de biais, sous la forme de vecteurs, intervenant dans le cumul des activations en entrée d'une unité, $W^{(1)}$ et $W^{(2)}$, les matrices de poids des connexions entre couches et Φ, la fonction d'activation de la couche cachée. Dans cette configuration, cette fonction est non linéaire, avec le choix dans notre de cas de $softsign(a) = |a|/(1+|a|)$ qui présente l'avantage de permettre des temps d'apprentissage réduits.

La différence entre le score ainsi calculé pour une séquence véritablement observée et une séquence altérée en changeant son mot central est utilisée comme critère d'optimisation pour la mise à jour à la fois des poids des connexions du réseau et de la valeur des dimensions de la représentation des mots de la séquence d'entrée. Cette mise à jour est réalisée de façon classique par l'application d'une descente de gradient stochastique. Le critère d'optimisation est donc un critère d'ordonnancement, à la différence des travaux antérieurs à (Collobert & Weston, 2008), qui optimisaient la log-vraisemblance pour le dernier mot de la séquence et devaient donc évaluer en sortie une probabilité pour tous les mots du vocabulaire pour chaque séquence en entrée. Plus formellement, ce critère d'ordonnancement spécifie que le score d'une séquence observée $\langle w_i \rangle$ doit être plus grand que celui de tout autre séquence $\langle \tilde{w_j} \rangle$ produite par remplacement du mot central de la séquence

observée par un autre mot du dictionnaire, et ceci avec une marge de μ, donc tel que $score(\langle w_i \rangle) > \mu + score(\langle \tilde{w_i} \rangle)$, où $\mu = 0,1$ comme dans (Collobert & Weston, 2008).

Au final, la représentation modifiée par ce critère de chaque mot de la séquence d'entrée est ensuite stockée pour être réutilisée lorsqu'une autre séquence contenant ce mot est présentée en entrée du réseau. Les représentations des mots sont ainsi adaptées de façon incrémentale en fonction du critère d'optimisation retenu.

3.3 Stratégie d'adaptation des représentations distribuées à la tâche

À la section précédente, nous avons défini la façon dont sont construites les représentations lexicales distribuées pour l'étiquetage en rôles événementiels, en reprenant fortement les principes définis dans (Collobert & Weston, 2008). Nous avons néanmoins réalisé une modification spécifique de cette méthode pour une meilleure adaptation des représentations construites à notre tâche. L'idée sous-jacente à cette modification est de favoriser, dans les exemples fournis pour l'entraînement du modèle, la présence de mots importants du domaine, de telle sorte que l'apprentissage s'effectue plus rapidement.

Pour évaluer l'importance d'un mot par rapport à un domaine, lequel s'identifie dans notre cas à un type d'événement, nous adoptons une approche faiblement supervisée en évaluant la proximité sémantique entre ce mot et un ensemble de mots représentatifs des événements considérés, appelés *étiquettes événements*. Par exemple, dans le cas du corpus MUC que nous avons utilisé pour nos expérimentations de la section 4, les événements sont des attaques terroristes et les mots choisis pour les représenter sont les étiquettes événements $\{attack, bombing, kidnapping, arson\}$. La proximité sémantique entre un mot et une étiquette événement est définie par la mesure de *Leacock Chodorow*. Cette mesure de similarité lexicale se fonde sur WordNet, en l'occurrence sa version 3.0, et dépend de la longueur du chemin le plus court entre deux synsets dans la hiérarchie de WordNet, normalisée par la hauteur de cette hiérarchie. Plus formellement, elle s'écrit : $-log(p/2 \cdot D)$ où p est la longueur du chemin entre les synsets des mots considérés et D est la hauteur de la hiérarchie de WordNet. La proximité d'un mot par rapport à un domaine est ainsi donnée par la valeur moyenne de la mesure de *Leacock Chodorow* entre ce mot et chacun des mots événements du domaine.

Pour favoriser la présence des mots importants du domaine dans les exemples, nous choisissons de modifier la stratégie de sélection du mot remplaçant le mot central d'une séquence exemple lors de la corruption de cette séquence. Dans (Collobert & Weston, 2008), ce choix est aléatoire parmi la totalité des mots du vocabulaire pris en compte pour construire les représentations. Dans notre cas, nous utilisons la méthode d'évaluation de l'importance d'un mot par rapport au domaine présentée ci-dessus pour ordonner les mots du vocabulaire et choisir le mot remplaçant de façon aléatoire parmi les mots ayant un score supérieur à un seuil donné.

3.4 Utilisation des représentations distribuées pour l'étiquetage en rôles événementiels

Les représentations apprises pour chacun des mots permettent de calculer les traits des exemples donnés en entrée du classifieur supervisé en vue de prédire leur étiquette, *i.e.* leur rôle événementiel. Néanmoins, les rôles événementiels ne sont en général pas occupés par de simples mots mais plutôt par des groupes nominaux, pouvant s'identifier dans certains cas à des entités nommées. Pour l'identification de ces rôles événementiels, nous avons donc opéré en deux temps. En premier lieu, nous avons appliqué un analyseur en chunks pour identifier les candidats à ces rôles. En l'occurrence, tout chunk nominal est considéré comme un candidat, les constituants des autres chunks recevant une étiquette NULL (cf. section 4.2).

Dans un second temps, nous avons appliqué un classifieur préalablement entraîné sur un corpus annoté pour décider quel rôle, s'il en occupe un, un chunk occupe pour le type d'événement considéré. Pour ce faire, il est nécessaire de passer de la représentation construite pour chaque mot à la représentation d'un chunk. Ce passage est réalisé via le mécanisme du *max-pooling*. Un chunk de N mots est ainsi représenté avec le même nombre de dimensions qu'un mot et chacune de ses dimensions i prend pour valeur $max(w_{i1}, ..., w_{iN})$ où w_{ij} est la valeur de la dimension i pour le mot w_j constituant le chunk.

Pour la classification proprement dite, nous nous appuyons sur la variante *Extra-Trees* (Geurts *et al.*, 2006) des forêts d'arbres décisionnels telle qu'elle est implémentée dans (Pedregosa *et al.*, 2011).

4 Expérimentations et résultats

4.1 Description de la tâche

Nous avons évalué le système présenté sur les données de la campagne d'évaluation MUC-4, qui forment un corpus d'évaluation standard pour la tâche d'extraction d'événement. Le corpus d'entraînement comporte 1 500 textes et modèles d'événements (*template*) instanciés associés. La tâche consiste à extraire les informations descriptives d'événements terroristes en Amérique Latine. Étant donné un texte, il s'agit de remplir une structure pour chaque événement décrit (par exemple attaque, enlèvement, prise d'otage, pose de bombe, etc.). Si le texte décrit plus d'un événement, il faut remplir une structure pour chacun d'eux. Les tests officiels, nommés TST3 et TST4, contiennent 100 documents chacun provenant de cet ensemble. Nous avons entraîné notre système sur 1300 documents et l'avons testé à chaque fois sur le même ensemble de test, formé de la conjonction des deux ensembles de tests TST3+TST4.

Une modèle d'événement comporte un ensemble d'attributs prédéfinis correspondant aux valeurs qui doivent être trouvées dans les textes, (dans les modèles d'événements de MUC-4, il y a 25 attributs). Ces attributs sont de types différents qui nécessitent d'être traités différemment, les valeurs de ces attributs devant être extraites ou inférées à partir des textes [3]. Ces attributs peuvent être divisés en trois catégories :

1. les attributs de type texte : ces attributs sont remplis par des chaînes de caractères extraites directement des textes (6. INCIDENT : INSTRUMENT ID, 9. PERP : INDIVIDUAL ID, 10. PERP : ORGANIZATION ID, 12. PHYS TGT : ID, 18. HUM TGT : NAME, 19. HUM TGT : DESCRIPTION, 6. INCIDENT : INSTRUMENT ID, 7. INCIDENT : INSTRUMENT TYPE). Ils ne correspondent pas forcément à une entité nommée ;
2. les attributs calculés : les valeurs doivent être calculées à partir d'extraits de textes. Par exemple, INCIDENT : DATE doit être inférée d'expressions temporelles telles que *today, last week,* etc.
3. les attributs à valeur contrainte : la valeur de ce type d'attribut provient d'un ensemble fini de valeurs possibles. Elles doivent souvent être inférées des documents.

Pour notre évaluation, nous nous sommes concentrés sur l'instanciation des attributs texte, de façon similaire aux autres systèmes de l'état de l'art. De façon similaire à (Patwardhan & Riloff, 2009), nous distinguons cinq grands groupes d'attributs :

AutInd	(PERP :INDIVIDUAL ID)	*AutOrg*	(PERP :ORGANIZATION ID)
Cible	(PHYS TGT :ID)	*Victime*	(HUM TGT :NAME, HUM TGT :DESCRIPTION)
Arme	(INCIDENT :INSTRUMENT ID, INCIDENT :INSTRUMENT TYPE)		

Nous évaluons ensuite la précision des extractions. Notons que, comme dans les travaux comparables (Patwardhan & Riloff, 2009; Huang & Riloff, 2010, 2011, 2012b), nous ne nous intéressons pas à la construction complète des structures événementielles mais seulement à l'identification des rôles événementiels, quel que soit l'événement auquel ils sont reliés. Pour établir la correspondance entre les valeurs extraites et les valeurs de référence, on compare les têtes des chunks (l'extraction de *men* est considérée correcte pour une réponse attendue de *five armed men*), et on fusionne les extractions multiples (de sorte que plusieurs chunks extraits partageant la même tête ne sont comptés qu'une seule fois). Enfin, cette évaluation prend en compte les rôles multi-valués, en distinguant les conjonctions (lorsque plusieurs victimes sont nommées, on doit les trouver toutes) et les disjonctions (lorsque la même entité a plusieurs noms, il suffit d'en trouver un seul).

4.2 Étiquetage du corpus pour l'apprentissage supervisé

Comme indiqué dans la section 3.4, nous utilisons un classifieur permettant d'associer chaque chunk du texte à un rôle événementiel. Ce classifieur doit donc être entraîné sur un corpus annoté correspondant à cette tâche, qui est construit automatiquement à partir des événements de référence. Les valeurs des attributs sont retrouvées dans les documents correspondants, en appliquant un seuil de distance minimale pour aligner les mots. Par exemple, si un attribut a trois valeurs possibles, chacune étant formée de plusieurs mots, nous recherchons tous les mots dans le texte. Si les positions des différents mots composant une valeur sont suffisamment proches, on attribue l'étiquette du rôle à l'empan délimité par ces mots.

3. A l'exception d'attributs de méta-données comme les attributs 0 (MESSAGE :ID) et 1 (MESSAGE : TEMPLATE).

Cet étiquetage automatique a été réalisé en fonction des syntagmes (*i.e.* les chunks) proposés par l'outil SENNA ("Semantic/syntactic Extraction using a Neural Network Architecture", (Collobert *et al.*, 2011)), où à chaque mot est attribué un tag unique, soit mot simple (S-NP), début de chunk (B-NP), interne à un chunk (I-NP) ou fin de chunk (E-NP). Nous associons à l'attribut le plus petit chunk englobant sa valeur. Toutes les variantes des modèles instanciés sont prises en compte. Les groupes restants, qui ne couvrent aucune valeur d'attribut, sont associés à une étiquette NULL. Un exemple de phrase annotée de cette façon est fourni ci-dessous.

phrase initiale	Guerrillas	attacked	the	Santo	Tomas	presidential	farm
chunks (SENNA)	S-NP	S-VP	B-NP	I-NP	I-NP	I-NP	E-NP
rôles événementiels	S-AGENT	NULL	B-TARGET	I-TARGET	I-TARGET	I-TARGET	E-TARGET

4.3 Expérimentations

Après l'annotation automatique du corpus et une normalisation de base (passage du corpus en minuscules, suppression des espaces en trop, découpage en phrases), les représentations lexicales sont apprises en appliquant le réseau de neurones présenté à la figure 1.

Après expérimentations, nous avons retenu des représentations lexicales formées par des vecteurs à 50 dimensions obtenus par application sur des séquences de 5 mots, dénommés *DRVR-50* (pour *Domain-Relevant Vector Representations*. Comme indiqué à la section 3.2, nous avons utilisé le réseau de neurones avec *softsign* comme fonction non linéaire. Vu la faible complexité de cette fonction, la durée d'entraînement est rapide (environ 12 heures), en comparaison des semaines mentionnées dans (Turian *et al.*, 2010).

Nous avons effectué un certain nombre d'expérimentations, que nous ne détaillerons pas ici, afin de déterminer la meilleure combinaison des paramètres importants de notre système. Parmi ceux-ci, nous avons accordé une attention toute particulière à la méthode de corruption des séquences en considérant trois conditions de choix aléatoire du mot corrupteur : choix parmi tout le vocabulaire, choix parmi les mots les plus fréquents et choix parmi les mots les plus liés au domaine selon le critère présenté à la section 3.3. Les meilleurs résultats ont été obtenus avec cette dernière condition, montrant ainsi l'intérêt du mécanisme d'adaptation faiblement supervisé au domaine que nous avons proposé pour la construction

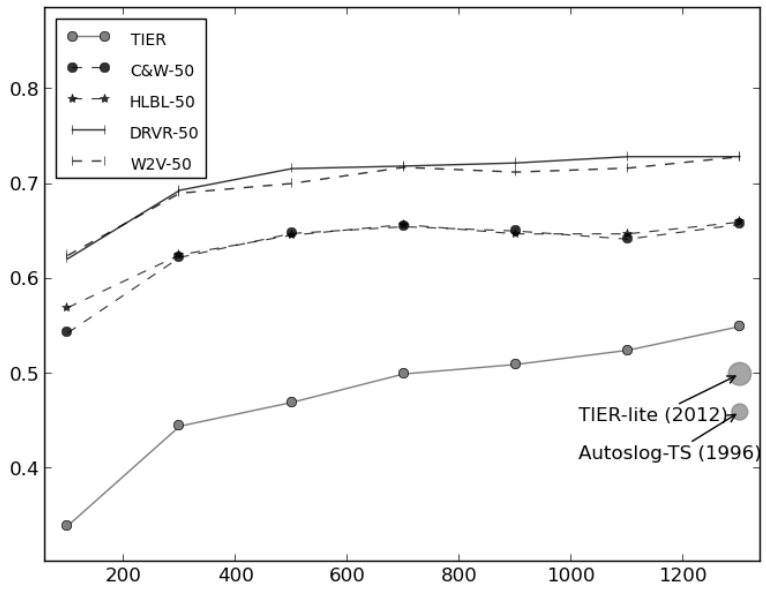


FIGURE 2 – F1-Mesure pour les rôles texte de TST3+TST4 avec différents paramètres, en relation avec la courbe d'apprentissage de TIER (Huang & Riloff, 2012a). Les points gris représentent des résultats marquant de la tâche.

TST3 + TST4						
Approches faiblement supervisées						
	AutInd	**AutOrg**	**Cible**	**Victime**	**Arme**	**Moyenne**
Autoslog-TS (1996)	33/49/40	53/33/41	54/59/56	49/54/51	38/44/41	45/48/46
Piper$_{Best}$ (2007)	39/48/43	55/31/40	37/60/46	44/46/45	47/47/47	44/36/40
TIER$_{lite}$ (2012)	47/51/47	60/39/47	37/65/47	39/53/45	53/55/54	47/53/**50**
Chambers+Jurafsky (2011)	–	–	–	–	–	44/36/40
Modèles supervisés						
GLACIER (2009)	51/58/54	34/45/38	43/72/53	55/58/56	57/53/55	48/57/52
TIER (2011)	48/57/52	46/53/50	51/73/60	56/60/58	53/64/58	51/62/56
(Huang & Riloff, 2012b)	54/57/56	55/49/51	55/68/61	63/59/61	62/64/63	58/60/**59**
Modèles neuronaux						
C&W-50	80/55/65	64/65/64	76/72/74	53/63/57	85/64/73	68/63/65
HLBL-50	81/53/64	63/67/65	78/72/75	53/63/58	93/64/75	69/62/66
W2V-50	79/57/66	88/71/79	74/72/73	69/75/71	97/65/78	77/68/72
DRVR-50	79/57/66	91/74/81	79/57/66	77/75/76	92/58/81	80/67/**73**

TABLE 1 – Résultats sur les rôles texte de TST3 + TST4 P/R/F1 (Précision/Rappel/F1-Mesure)

des représentations lexicales distribuées.

Pour la tâche d'étiquetage supervisé, nous utilisons comme indiqué précédemment un algorithme de forêts d'arbres décisionnels (Extra-Trees), avec 500 arbres, valeur adoptée pour toutes les expérimentations menées.

Les résultats présentés à la figure 2 sont calculés pour les rôles considérés (AutInd, AutOrg, Cible, Victime, Arme). Nous observons que nos représentations lexicales (*DRVR-50*) surpassent les résultats de l'état de l'art indiqués par les points gris, ce qui montre qu'elles permettent de représenter des informations sémantiques au moins équivalentes pour la tâche sans avoir à ajouter d'autres traits. Par ailleurs, on peut voir qu'elles présentent une bonne stabilité par rapport à la taille du corpus d'apprentissage. La méthode que nous proposons est donc une piste intéressante pour développer rapidement des systèmes d'extraction d'événements sur un nouveau domaine avec peu de données annotées.

La table 1 présente des résultats comparatifs plus détaillés. On peut voir dans ce tableau que nos résultats surpassent ceux des modèles faiblement supervisés (0,73 vs 0,59) et supervisés (0,73 vs 0,56). Les rôles *AutOrg* et *Arme* obtiennent même une très bonne précision, ce qui signifie que pour ces rôles, un filtre supplémentaire pour éliminer les faux candidats n'est pas nécessaire. De façon générale, le compromis entre la justesse de la réponse et le nombre de candidats trouvés devra être étudié plus précisément dans de futurs travaux.

L'aspect innovant de notre système concerne l'utilisation des représentations lexicales apprises par un modèle neuronal. Pour étudier de façon plus poussée l'influence de ces représentations, nous avons comparé nos représentations apprises sur le corpus MUC-4 selon le modèle détaillé en section 3 avec des modèles de représentations lexicales existants : nous avons utilisé les données mises à disposition par (Turian *et al.*, 2010), et plus précisément, selon les modèles de C&W et HLBL [4]. Ces représentations sont construites à partir d'un corpus plus important et plus généraliste d'articles de journaux (corpus Reuters RCV1). Les résultats obtenus avec ces représentations lexicales sont reportés dans la table 1 et montrent que les scores obtenus avec notre modèle restent supérieurs (avec une F1-mesure de 0,72 contre 0,65 et 0,66, due surtout à une meilleure précision). Ceci met en évidence qu'un modèle appris sur un corpus spécifique à un domaine permet d'obtenir de meilleurs résultats, même si ce corpus est de taille beaucoup moins importante (alors qu'il est d'usage de considérer que les modèles neuronaux nécessitent souvent des données d'entraînement importantes).

De façon complémentaire, nous avons comparé notre méthode pour apprendre les représentations lexicales sur le corpus MUC-4 avec la méthode proposée par (Mikolov *et al.*, 2011) [5], en utilisant le même corpus. Les résultats sont présentés dans la table 1 sous le nom *W2V50*. Les résultats obtenus sont alors comparables avec ceux obtenus par notre système (légèrement moins bons), ce qui confirme que l'utilisation d'un corpus spécifique au domaine considéré est bien un atout intéressant.

4. Ces données sont disponibles sur `http://metaoptimize.com/projects/wordreprs`.
5. Son code pour générer les représentations est disponible à l'adresse : `https://code.google.com/p/word2vec`.

5 Conclusions et perspectives

Nous avons présenté une nouvelle approche d'étiquetage en rôles événementiels qui permet de réduire le nombre de traits à concevoir manuellement en utilisant des représentations lexicales distribuées apprises de manière non supervisée. Ces types de représentation sont connus pour être indépendants de la tâche, et nous avons montré que l'on pouvait les utiliser dans une tâche d'extraction d'événements, en obtenant des résultats qui surpassent les résultats actuels sur la même tâche. De plus, les représentations apprises le sont en tenant compte du domaine à analyser et cela contribue à l'amélioration des résultats obtenus. Nous avons aussi montré qu'elles étaient stables sur différentes tailles de corpus d'apprentissage. Un second point important de ces résultats concerne l'adaptation d'un système à un nouveau domaine. Dans notre cas, il suffit de fournir seulement des exemples de valeurs de rôles à étiqueter et un corpus pas nécessairement très important, et on peut développer rapidement un système d'extraction d'information. Aucune définition de nouveaux traits ou étude de leur adaptation au domaine n'est requise. Il reste à vérifier que l'on peut obtenir d'aussi bons résultats sur un domaine différent.

Dans le futur, nous envisageons de tester d'autres architectures de réseau de neurones pour tirer parti d'informations que l'on peut obtenir à partir d'un analyseur tel qu'un analyseur à base de grammaire probabiliste hors-contexte. Nous envisageons aussi d'étendre le système à l'ensemble de la tâche, en considérant toutes les sous-tâches ensembles lors de l'apprentissage de manière à considérer les relations qu'elles entretiennent.

Références

BENGIO Y., DUCHARME R. & VINCENT P. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, **3**, 1137–1155.

BENGIO Y., SCHWENK H., SENÉCAL J.-S., MORIN F. & GAUVAIN J.-L. (2006). Neural probabilistic language models. In D. HOLMES & L. JAIN, Eds., *Innovations in Machine Learning*, volume 194 of *Studies in Fuzziness and Soft Computing*, p. 138–186. Springer Berlin Heidelberg.

BUNESCU R. & MOONEY R. J. (2004). Collective information extraction with relational markov networks. In *42nd Annual Meeting of the Association for Computational Linguistics (ACL'04)*, p. 438–445.

CHIEU H. L., NG H. T. & LEE Y. K. (2003). Closing the gap : Learning-based information extraction rivaling knowledge-engineering methods. In *41st Annual Meeting of the Association for Computational Linguistics (ACL'03)*, p. 216–223.

CHUN H.-W., HWANG Y.-S. & RIM H.-C. (2005). Unsupervised event extraction from biomedical literature using co-occurrence information and basic patterns. In *IJCNLP 2004*, p. 777–786. Springer.

COHEN K. B., VERSPOOR K., JOHNSON H. L., ROEDER C., OGREN P. V., BAUMGARTNER JR W. A., WHITE E., TIPNEY H. & HUNTER L. (2009). High-precision biological event extraction with a concept recognizer. In *Workshop on Current Trends in Biomedical Natural Language Processing : Shared Task*, p. 50–58.

COLLOBERT R. & WESTON J. (2008). A unified architecture for natural language processing : Deep neural networks with multitask learning. In *25rd International Conference of Machine learning*, p. 160–167 : ACM.

COLLOBERT R., WESTON J., BATTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Approach*, **12**, 2493–2537.

FREITAG D. (1998). Information extraction from HTML : Application of a general machine learning approach. In *AAAI*, p. 517–523.

GEURTS P., ERNST D. & WEHENKEL L. (2006). Extremely randomized trees. *Machine Learning*, **63**(1), 3–42.

GRISHMAN R. & SUNDHEIM B. (1996). Message understanding conference-6 : A brief history. In *COLING 1996*, p. 466–471.

HOBBS J. R., APPELT D., TYSON M., BEAR J. & ISRAEL D. (1992). SRI International : Description of the FASTUS system used for MUC-4. In *4th Conference on Message understanding*, p. 268–275.

HUANG R. & RILOFF E. (2010). Inducing domain-specific semantic class taggers from (almost) nothing. In *48th Annual Meeting of the Association for Computational Linguistics*, p. 275–285.

HUANG R. & RILOFF E. (2011). Peeling back the layers : Detecting event role fillers in secondary contexts. In *ACL 2011*, p. 1137–1147.

HUANG R. & RILOFF E. (2012a). Bootstrapped training of event extraction classifiers. In *13th Conference of the European Chapter of the Association for Computational Linguistics*, p. 286–295.

HUANG R. & RILOFF E. (2012b). Modeling textual cohesion for event extraction. In *26th Conference on Artificial Intelligence (AAAI 2012)*.

HUNG S.-H., LIN C.-H. & HONG J.-S. (2010). Web mining for event-based commonsense knowledge using lexico-syntactic pattern matching and semantic role labeling. *Expert Systems with Applications*, **37**(1), 341–347.

JEAN-LOUIS L., BESANÇON R. & FERRET O. (2011). Text segmentation and graph-based method for template filling in information extraction. In *5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*, p. 723–731.

KRUPKA G., JACOBS P., RAU L. & IWAŃSKA L. (1991). GE : Description of the NLToolset System as Used for MUC-3. In *3rd Conference on Message understanding*, p. 144–149.

LANDAUER T. K., FOLTZ P. W. & LAHAM D. (1998). An introduction to latent semantic analysis. *Discourse processes*, **25**(2-3), 259–284.

LEHNERT W., CARDIE C., FISHER D., MCCARTHY J., RILOFF E. & SODERLAND S. (1992). University of Massachusetts : MUC-4 test results and analysis. In *4th Conference on Message understanding*, p. 151–158.

MICHAELY R., THALER R. H. & WOMACK K. L. (1995). Price reactions to dividend initiations and omissions : overreaction or drift ? *The Journal of Finance*, **50**(2), 573–608.

MIKOLOV T., KOMBRINK S., BURGET L., CERNOCKY J. & KHUDANPUR S. (2011). Extensions of recurrent neural network language model. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 5528–5531.

PATWARDHAN S. (2010). *Widening the field of view of information extraction through sentential event recognition*. PhD thesis, University of Utah.

PATWARDHAN S. & RILOFF E. (2007). Effective information extraction with semantic affinity patterns and relevant regions. In *EMNLP-CoNLL 2007*, p. 717–727.

PATWARDHAN S. & RILOFF E. (2009). A unified model of phrasal and sentential evidence for information extraction. In *2009 Conference on Empirical Methods in Natural Language Processing*, p. 151–160.

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. & DUCHESNAY E. (2011). Scikit-learn : Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

RILOFF E. (1996a). Automatically generating extraction patterns from untagged text. In *AAAI'96*, p. 1044–1049.

RILOFF E. (1996b). An empirical study of automated dictionary construction for information extraction in three domains. *Artificial intelligence*, **85**(1), 101–134.

STRASSEL S., PRZYBOCKI M. A., PETERSON K., SONG Z. & MAEDA K. (2008). Linguistic resources and evaluation techniques for evaluation of cross-document automatic content extraction. In *LREC 2008*.

SUDO K., SEKINE S. & GRISHMAN R. (2003). An improved extraction pattern representation model for automatic ie pattern acquisition. In *41st Annual Meeting of the Association for Computational Linguistics (ACL'03)*, p. 224–231.

SURDEANU M., TURMO J. & AGENO A. (2006). A hybrid approach for the acquisition of information extraction patterns. In *EACL-2006 Workshop on Adaptive Text Extraction and Mining (ATEM 2006)*, p. 48–55.

TURIAN J., RATINOV L. & BENGIO Y. (2010). Word representations : a simple and general method for semi-supervised learning. In *48th international Annual Meeting on Association for Computational Linguistics*, p. 384–394.

VOORHEES E. & TONG R. (2011). Overview of the TREC 2011 medical records track. In *TREC 2011*.

YAKUSHIJI A., TATEISI Y., MIYAO Y. & TSUJII J. (2001). Event extraction from biomedical papers using a full parser. In *Pacific Symposium on Biocomputing*, volume 6, p. 408–419.

YANGARBER R., GRISHMAN R., TAPANAINEN P. & HUTTUNEN S. (2000). Automatic acquisition of domain knowledge for information extraction. In *18th Conference on Computational linguistics (COLING 2000)*, p. 940–946.

Utilisation de représentations de mots pour l'étiquetage de rôles sémantiques suivant FrameNet

William Léchelle, Philippe Langlais
DIRO, Université de Montréal
{lechellw, felipe} @ iro.umontreal.ca

Résumé. D'après la sémantique des cadres de Fillmore, les mots prennent leur sens par rapport au contexte événementiel ou situationnel dans lequel ils s'inscrivent. FrameNet, une ressource lexicale pour l'anglais, définit environ 1000 cadres conceptuels couvrant l'essentiel des contextes possibles.
Dans un cadre conceptuel, un prédicat appelle des arguments pour remplir les différents rôles sémantiques associés au cadre. Nous cherchons à annoter automatiquement ces rôles sémantiques, étant donné le cadre sémantique et le prédicat, à l'aide de modèles à maximum d'entropie.
Nous montrons que l'utilisation de représentations distribuées de mots pour situer sémantiquement les arguments apporte une information complémentaire au modèle, et améliore notamment l'étiquetage de cadres avec peu d'exemples d'entrainement.

Abstract. According to Frame Semantics (Fillmore 1976), words' meaning are best understood considering the semantic frame they play a role in, for the frame is what gives them context. FrameNet defines about 1000 such semantic frames, along with the roles arguments can fill in this frame. Our task is to automatically label arguments' roles, given their span, the frame, and the predicate, using maximum entropy models.
We make use of distributed word representations to improve generalisation over the few training exemples available for each frame.

Mots-clés : rôles sémantiques ; représentations distribuées ; maximum d'entropie.

Keywords: semantic role labelling ; distributed word representations.

1 Introduction

Développé depuis 1997 à l'université Berkeley, le projet FrameNet [1] définit un peu plus de 1000 cadres sémantiques, visant à couvrir tous les évènements ou tous les contextes possibles, au niveau le plus général. Des rôles sémantiques sont définis par chaque cadre, et ces rôles qui seront remplis dans le texte par des arguments (segments de phrase). Les différents cadres sémantiques sont reliés par des relations de cadre à cadre (par exemple, un cadre peut être un sous-cas d'un autre plus général), pour former une hiérarchie. Par exemple, le cadre `Communication` est décrit comme suit (les rôles sont mis en évidence) :

> Un `Communicateur` transmet un `Message` à un `Destinataire` ; le `Sujet` et `Medium` de la communication pouvant aussi être exprimés. Ce cadre ne spécifie pas la méthode de communication (oral, écrit, geste, etc.). Les cadres qui héritent de ce cadre général de `Communication` peuvent ajouter des détails au `Medium` de différentes façons (en français, à la radio, dans une lettre), ou à la `Façon` de communiquer (bavardage, diatribe, cri, murmure).

La figure 1 montre un exemple d'annotation des différents cadres présents dans une phrase, avec les arguments remplissant les rôles exprimés dans ce cadre.

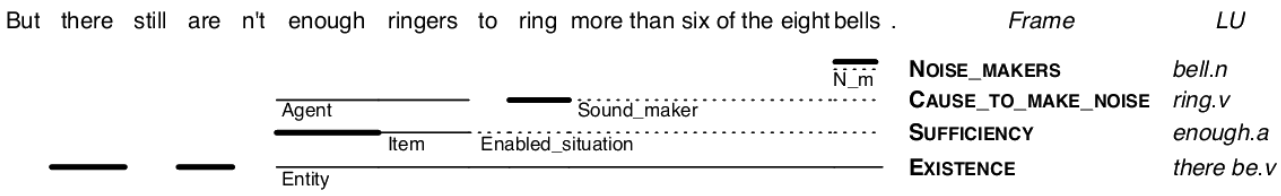


FIGURE 1 – Exemple d'annotation de tous les cadres présents. Le cadre `Cause_de_bruit` (`Cause to make noise`) est appelé par l'unité lexicale *ring.v*, la cible. Dans ce cadre, les rôles `Agent` et `Producteur_de_son` sont annotés, remplis par *enough ringers* et *more than six of the eight bells* respectivement. Les cadres `Faiseur_de_bruit`, `Suffisance` et `Existence` sont également annotés dans la même phrase. Figure tirée de Das *et al.* 2010.

L'annotation sémantique automatique se déroule généralement en une succession d'étapes (par exemple dans Das *et al.* (2010) et Punyakanok *et al.* (2008)) :

- Identifier les prédicats (dits mots "cibles") qui appellent des cadres. Dans la figure 1, les cibles (*bells*, *ring*, *enough*, et *there are*) sont soulignées en gras.
- Désambiguiser le cadre appelé par chaque cible (`Noise_makers`, `Cause_to_make_noise`, etc.)
- Déterminer la position des arguments qui remplissent les rôles sémantiques (ou déterminer, pour un syntagme candidat, s'il est un argument ou non).
- **Étiqueter chacun des arguments avec le rôle qu'il remplit.**

Nous nous sommes concentrés sur cette dernière étape. Pour chaque occurence d'un cadre dans une phrase, on considère donc connus :

- la cible ;
- le cadre précis évoqué par la cible ;
- la position de chacun des arguments.

2 Données et évaluation

FrameNet fournit 2 ensembles de données [2]. D'une part, environ 170 000 phrases sont extraites du *British National Corpus* pour exemplifier l'usage de chaque cadre sémantique. Dans ce corpus, un seul cadre par phrase est annoté. D'autre part, environ 6000 phrases (provenant de 78 documents) sont complètement annotées, avec plusieurs cadres par phrase, et totalisent environ 24 000 instances de cadres. Ce deuxième corpus est plus représentatif du texte qu'aurait à traiter une application concrète et est plus adapté à l'entrainement de systèmes automatiques, mais est malheureusement beaucoup plus restreint.

1. `https://framenet.icsi.berkeley.edu/fndrupal/`
2. Nous utilisons la version 1.5, publiée en septembre 2010, téléchargée en février 2013.

Nous avons utilisé les deux corpus de données de FrameNet, le texte complètement annoté, et une partie des phrases exemples [3]. Au total, l'ensemble d'entrainement comporte 106 926 cadres, avec 1,45 argument par cadre.

Un cadre sémantique qui apparaît dans le jeu de test apparaît en moyenne 300 fois dans l'ensemble d'entrainement (appelé par différents prédicats, avec différents arguments). En pratique, certains cadres sont beaucoup plus représentés que d'autres, ce sur quoi on reviendra en section 5.4.

L'ensemble de test est le même que celui de (Das & Smith, 2011), soit 23 documents complètement annotés. Il comporte 4456 instances de cadres, soit 7209 arguments à classifier (1,6 argument par cadre en moyenne).

Pour permettre une certaine comparaison à l'état de l'art, la performance rapportée pour chacune des méthodes évaluées est la micro-précision sur cet ensemble d'arguments, c'est-à-dire la proportion d'arguments dont le rôle est correctement prédit par le modèle, tous cadres confondus. Une autre mesure de performance pertinente est la macro-précision, au sens de la performance moyenne de chacun des cadres (voir section 5.4).

Formellement, si on note n(f) le nombre d'arguments de l'ensemble de test à annoter pour le cadre f, et c(f) le nombre d'arguments annotés correctement par le modèle du cadre f, $p(f) = \frac{c(f)}{n(f)}$ est la proportion d'arguments annotés correctement pour le cadre f. Nos métriques s'écrivent alors :

$$\text{micro-précision} = \frac{\sum_{f \in \text{cadres}} c(f)}{\sum_{f \in \text{cadres}} n(f)}$$

$$\text{macro-précision} = \underset{f \in cadres}{\text{moyenne}}(\ p(f)\ ) = \frac{\sum_{f \in \text{cadres}} \frac{c(f)}{n(f)}}{\text{nb cadres}}$$

3 État de l'art

Plusieurs auteurs se sont attachés à identifier les arguments et leurs rôles pour les cadres sémantiques de FrameNet. Les premiers, Gildea & Jurafsky (2000) utilisent une cascade de modèles (*backoff*) s'appuyant sur les comptes de caractéristiques des noeuds de l'arbre syntaxique correspondant aux arguments. Leur modèle final obtient 76.9% de précision sur leur ensemble de test, pour la tâche d'étiquetage des rôles. Leurs données étaient les phrases exemples d'une version préliminaire de FrameNet comportant 67 cadres, soit 49 013 phrases annotées avec un cadre par phrase.

De leur travail ressort un équilibre entre la couverture de chaque modèle et sa précision : le modèle n'utilisant que la cible couvre 100% des cas mais est seulement précis à 41%. À l'inverse, les mots de tête constituent une caractéristique des plus fiables pour déterminer le rôle d'un argument : le modèle utilisant uniquement la cible et le mot de tête pour classifier un rôle obtient 86,7% de précision, mais ne couvre que 56% des données. Pour augmenter la couverture de ce modèle, les auteurs font une expérience pour grouper les noms du lexique (avec la technique de *clustering* décrite dans Lin 1998), ce qui leur permet d'obtenir un modèle couvrant 98% des mots de tête nominaux, et précis à 79.7%. Finalement, cela ajoute 0,8% de précision à leur modèle global, sur l'ensemble de développement.

Notre étude vise à être la continuation de cette expérience, en utilisant les représentations distribuées de mots pour généraliser les mots du lexique.

Cette idée a été employée récemment dans des expériences sur le FrameNet suédois naissant : Johansson *et al.* (2012) utilisent le clustering lexical de Brown (Brown *et al.* 1992) pour augmenter la couverture de caractéristiques lexicales (pour la classification de rôles), et rapportent une légère amélioration de performance.

En 2003, Fleischman *et al.* ont été les premiers à utiliser des modèles à maximum d'entropie pour l'identification du rôle des arguments, en obtenant des résultats proches de ceux de Gildea et Jurafsky (76% de précision avec des données obtenues automatiquement), sur des données comparables (40 000 phrases exemples de FrameNet).

SEMAFOR (Das *et al.* 2010), développé depuis 2010 à l'université Carnegie-Mellon, est un système complet d'annotation sémantique automatique. Pour résoudre notre tâche, ce système détermine, pour chaque rôle sémantique, l'emplacement

3. Nous n'avons considéré que les arguments constitués d'un seul mot.

de l'argument qui le comble (ou aucun argument si le rôle n'est pas exprimé). Ses prédictions sont précises (88%), mais avec un plus faible rappel (75%), c'est-à-dire que le modèle estime que certains rôles ne sont pas exprimés, alors qu'ils le sont. Notre système suit la méthode plus courante (de Johansson & Nugues 2007) qui consiste à détecter d'abord les arguments, pour ensuite les classifier en leurs rôles respectifs.

Étant donnés la cible, le cadre, et les positions exactes des arguments, SEMAFOR obtient un score F_1 de 81%. Leurs expériences utilisent la version 1.3 de FrameNet, semblable à nos propres données.

4 Représentations de mots

D'après Turian *et al.* 2010, l'utilisation de représentations distribuées de mots – apprises de manière non supervisée – est une méthode simple et générale pour améliorer la précision de systèmes d'apprentissage supervisé pour le traitement des langues.

Nous avons utilisé les représentations de mots fournies par Ronan Collobert, apprises par SENNA [4] (Collobert *et al.* 2011) à partir d'un grand corpus de texte non étiqueté (provenant essentiellement de Wikipedia). Cette ressource fournit la représentation par un vecteur de valeurs réelles, dans un espace de dimension 50, de 130 000 mots, les plus fréquents dans le corpus. On utilise la distance euclidienne pour mesurer la proximité de mots dans cet espace.

Pour donner une idée, la table 2 montre les plus proches voisins de quelques mots pris au hasard.

TABLE 2 – Mots les plus proches d'exemples choisis au hasard, d'après leur représentation vectorielle.

Exemple	plus proches voisins
characteristics	traits indicators measurements phenotypes ...
chalkboard	sofa washroom hallway bathroom darkroom skit ...
charlene	cynthia cathy benji angie ronnie julie caitlin cheryl ...
delighted	amazed thrilled dismayed ridiculed astonished ...
deregulate	liberalise reallocate unsettle penalise unnerve ...
falsifiability	teleology rationality holism causality ...
memorizing	deciphering interpreting embodying unlocking ...
parrots	wasps cormorants beetles lizards newts ...
planet	earth universe portal basestar mothership galaxy ...
retirement	tenure graduation incarceration signing ...
visible	confusing common hidden standing peculiar ...

5 Expériences

Dans l'idée, si des mots sont proches dans l'espace des représentations, c'est qu'ils sont sémantiquement proches – synonymes au sens large, typiquement. L'apprentissage devrait pouvoir profiter de cette connaissance pour annoter des exemples de test inconnus, proches sémantiquement d'exemples d'entrainement connus, avec le même rôle que ces derniers. Autrement dit, généraliser la connaissance des exemples d'entrainement aux arguments qui en sont sémantiquement proches.

5.1 Modèle de référence

Comme système de référence, nous avons entrainé des modèles à maximum d'entropie, un par cadre (les rôles que peuvent remplir les arguments sont différents pour chaque cadre).

Notre implémentation utilise Python et NLTK [5]. L'entrainement du modèle utilise l'algorithme du gradient conjugué ('CG').

4. http://ronan.collobert.com/senna/
5. http://nltk.org/

Le modèle s'appuie sur une quinzaine de caractéristiques de surface de l'argument. Avec l'exemple de l'`Agent` dans le cadre `Cause_de_bruit` (figure 1), les caractéristiques employées sont regroupées en catégories et présentées dans la table 3.

TABLE 3 – Caractéristiques du modèle de référence, ainsi que leur valeur dans l'exemple présenté en figure 1.

Caractéristiques par catégorie	Valeur dans l'exemple
Caractéristiques de base	
le texte de l'argument	`enough ringers`
le texte de la cible	`ring`
la position (en caractères) de l'argument dans la phrase	23
Position relative	
est-ce que l'argument est avant ou après la cible	avant
si l'argument est à la même place que la cible	non
la distance (en mots) entre la cible et l'argument	1
Nombre de mots	
le nombre de mots de l'argument	2
le nombre de mots pleins [6] de l'argument	2
Contenu	
le premier mot de l'argument	`enough`
le premier mot plein de l'argument	`enough`
Parties du discours	
partie du discours du premier mot de l'argument	JJ (adjectif)
partie du discours du dernier mot de l'argument	NNS (nom pluriel)
la partie du discours majoritaire dans l'argument	JJ
Arguments précédents de cadre	
nombre d'arguments déjà étiquteés dans le cadre	0

Ce système de référence arrive à 80.0% de précision, à comparer aux 76% de précision de Fleischman & Hovy (2003), ou aux 80.97% de F_1-mesure de Das *et al.* (2010), sur d'autres versions des données de FrameNet.

Dans notre travail, l'utilisation de représentations de mots vient améliorer ce système de référence en apportant de l'information supplémentaire (obtenue par apprentissage non supervisé).

5.2 Plus proches voisins

On cherche à utiliser la proximité sémantique d'arguments pour prédire leur rôle. Pour mesurer la proximité et calculer des distances dans l'espace des représentations, il faut situer les arguments dans cet espace. Nous sommes partis du principe qu'un mot appartenant à chaque argument devait le représenter sémantiquement. 50% des arguments du jeu de test sont constitués de plusieurs mots.

Prenons un exemple, dans le cadre de l'`Engagement` (`Commitment` : un `Orateur` prend un engagement auprès d'un `Destinataire`). L'argument de test *l'ambassadeur iraquien* est inconnu à l'entrainement, mais peut être représenté par *ambassadeur*, proche des exemples connus *personne* ou *porte-parole*. Ces mots remplissent généralement le même rôle dans ce cadre : `Orateur`. On peut alors conclure que *l'ambassadeur iraquien* remplit aussi le rôle d'`Orateur`.

Comme représentants sémantiques des arguments, nous avons choisi d'utiliser leurs têtes syntaxiques, déterminées grâce au Stanford Parser (de Marneffe *et al.* 2006). Par exemple, l'argument *their ignorance which was based on prominent views* est représenté par *ignorance*, car toute la proposition subordonnée dépend (indirectement) de *based*, qui dépend de *ignorance*, et *their* dépend aussi d'*ignorance*. Dans le cadre d'un `Jugement` (fait par une personne, pouvant être positif ou négatif), cet argument est ensuite classifié comme `Celui_ou_ce_qui_est_jugé`.

D'autres approches seraient possibles pour représenter les arguments : Surdeanu *et al.* (2003) en particulier propose un choix de représentant sémantique plus élaboré et probablement plus adapté. En particulier, dans le cas de syntagmes prépositionnels, la préposition n'est pas nécéssairement sémantiquement significative.

6. de plus de 5 caractères

Chaque argument est situé à l'emplacement de son mot représentatif dans l'espace des représentations [7]. On peut alors prédire le rôle d'un argument avec le modèle des k plus proches voisins.

Le modèle du 1-plus-proche-voisin, qui prédit pour un argument le rôle du mot annoté le plus proche dans l'ensemble d'entrainement (pour le cadre considéré) obtient 70% de précision, expérimentalement. C'est assez remarquable pour un modèle aussi simple : pour comparaison, le modèle qui prédit pour un argument le rôle le plus fréquemment annoté à l'entrainement arrive seulement à 50% de précision.

Prendre en compte plusieurs voisins dans la prédiction du rôle d'un argument réduit le bruit dans les données et améliore nettement les performances. La figure 4 montre les résultats du modèle des plus proches voisins en fonction de k. Dans nos expériences, utiliser cette prédiction comme caractéristique unique d'un classifieur à maximum d'entropie est un peu meilleur [8] que de l'utiliser directement, et c'est ce que nous avons fait ici.

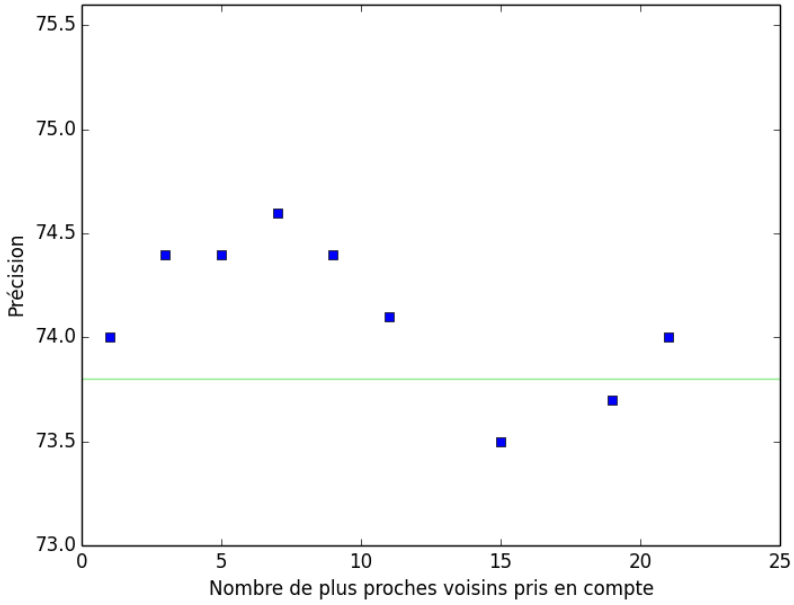


FIGURE 4 – Performance du modèle utilisant le rôle majoritaire parmi les k plus proches voisins de chaque argument comme seule caractéristique. La ligne représente la performance du modèle qui utiliserait uniquement la meilleure des caractéristiques du modèle de référence (la position relative de l'argument par rapport à la cible).

Pour intégrer la prédiction du modèle des plus proches voisins au modèle de référence, on peut simplement l'ajouter comme caractéristique d'un argument. La figure 5 montre les performances obtenues : avec plusieurs voisins, cette information améliore le modèle de référence (en vert), et permet d'arriver au niveau du système SEMAFOR, à l'état de l'art (en rouge pointillé), sur des données d'entrainement semblables.

5.3 Centres des exemples d'un rôle

Une autre façon de prédire le rôle d'un argument consiste à trouver quelle classe il représente le mieux : dans l'espace des représentations, on situe la position moyenne des mots représentant un rôle, et on assigne alors à un argument le rôle dont il est le plus proche.

Cela revient à partitionner l'espace des représentations en zones, une par rôle. Une zone est l'ensemble des points les plus proches du "représentant moyen" d'un rôle sémantique (comme un diagramme de Voronoï). On classifie alors les arguments en fonction de la zone dans laquelle ils se situent.

7. les mots de tête de 2% des arguments sont hors du vocabulaire des représentations, et alors seul le modèle de référence est utilisé
8. la différence est de l'ordre de 3-4%

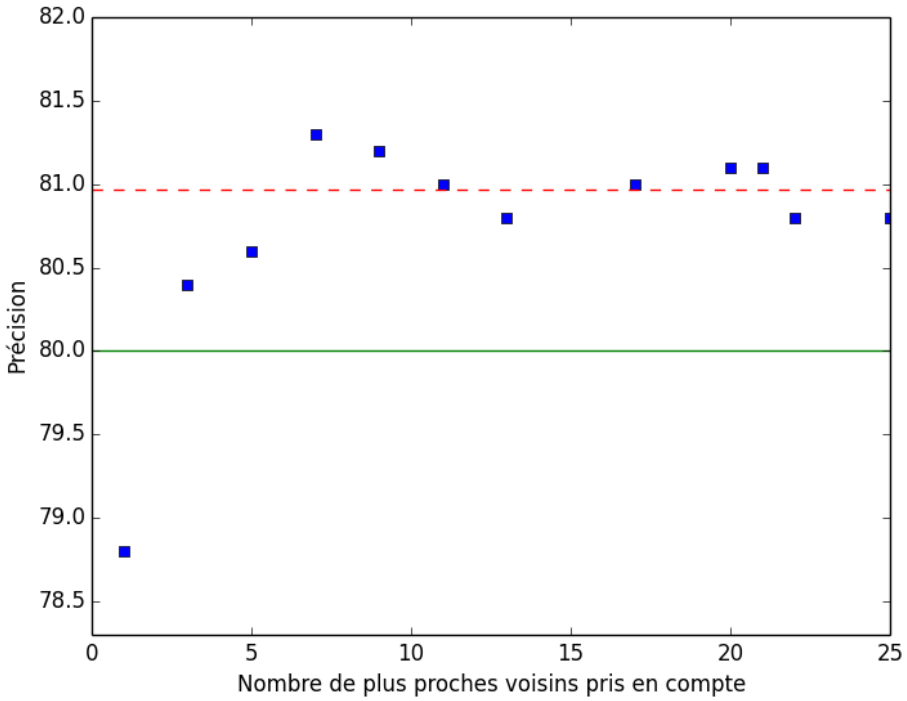


FIGURE 5 – Précision du modèle de référence informé de la prédiction du modèle des k plus proches voisins. La ligne verte représente le système de référence, et la ligne rouge pointillée SEMAFOR.

Ajoutée au modèle de référence, cette caractéristique permet d'atteindre 81.1% de précision (soit une amélioration de 1.1 point).

On peut combiner ce modèle avec celui des plus proches voisins, et ajouter les deux prédictions au modèle de référence. La figure 6 montre les résultats d'une telle combinaison. La meilleure performance est de 81.5%, avec 20 voisins.

5.4 Discussion

La table 7 récapitule les différents résultats, en termes de micro-précision (cf section 2). Rappelons que Fleischman & Hovy utilisent une version de FrameNet datant de 2002, c'est-à-dire sensiblement moins de données. Nous mesurons nos performances sur le même ensemble de test que SEMAFOR, en utilisant des données similaires à l'entrainement.

TABLE 7 – Récapitulatif des performances (micro-précision).

Modèle	Performance
k plus proches voisins	74.6%
Fleischman et Hovy (2003)	76%
Référence	80.0%
SEMAFOR (2010)	81.0%
Référence + centres	81.1%
Référence + plus proches voisins	81.3%
Référence + centres + PPV	81.5%

En regardant les résultats plus en détail, on observe que les gains en performance proviennent en large part des cadres avec le moins d'exemples. Il est alors intéressant de mesurer la macro-précision qu'obtiennent les différents modèles (cf section 2). Par rapport aux performances rapportées précédemment, c'est comme si la performance du modèle de chaque cadre n'était plus pondérée par le nombre d'exemples de ce cadre dans le jeu de test.

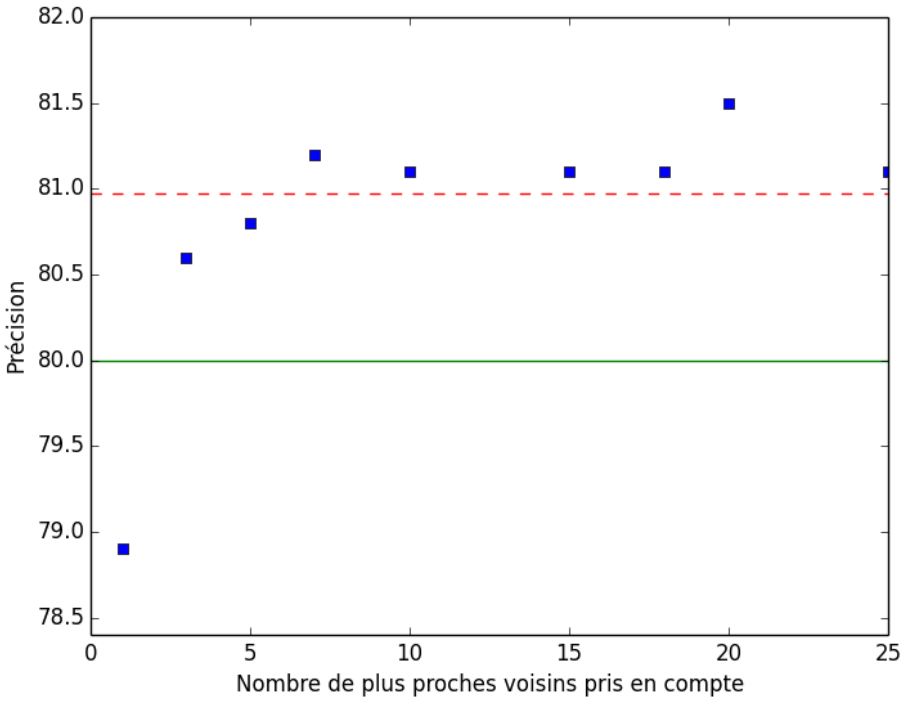


FIGURE 6 – Précision des modèles combinés de reférence, des plus proches voisins et des centres des rôles.

TABLE 8 – Macro-précision pour chacune des méthodes. Les cadres avec moins d'exemples prennent davantage d'importance, par rapport au calcul de la micro-précison.

Modèle	Précision moyenne des cadres
Centre des rôles le plus proche	69.6%
k plus proches voisins	70.0%
Référence	**73.2%**
Référence + centres	75.8%
Référence + plus proches voisins	75.2%
Référence + centres + PPV	76.1%

La table 8 montre la macro-précision des modèles que nous avons testés. Comme on peut voir, l'apport des représentations de mots est plus net avec cette mesure. En particulier, les 20% de cadres sémantiques avec le moins d'exemples d'entrainement améliorent leur précision moyenne de 5 points lorsque l'on rajoute la prédiction utilisant les centres des rôles au modèle de référence.

Cette observation est cohérente avec la vision évoquée plus haut, à savoir que l'abstraction sur les mots du lexique permet de mieux généraliser les données disponibles, surtout lorsqu'elles sont peu importantes (peu d'exemples par classe). Cette dernière mesure est d'autant plus pertinente que les cadres peu représentés dans l'ensemble de test sont aussi les cadres les plus difficiles à entrainer, du fait de leur plus faible nombre d'exemples d'entrainement.

6 Travaux futurs

L'implémentation des règles proposées dans Surdeanu *et al.* (2003) pour déterminer le mot le plus représentatif du contenu d'un argument serait une direction naturelle pour poursuivre ce travail. En particulier, les règles qui déterminent le mot de tête de syntagmes prépositionnels (la préposition) sont inadaptées à l'usage qu'on souhaite en faire ici, ce que Surdeanu propose d'améliorer.

Du point de vue de l'apprentissage, on pourrait améliorer l'entrainement des modèles. Actuellement, les performances globales sont peu sensibles à la variation du nombre de voisins pris en compte, passé un certain stade (voir figure 5). Un ensemble de développement permettrait de mieux adapter les paramètres à chaque méthode, et en particulier de faire varier la valeur du nombre de voisins considérés en fonction du cadre sémantique et des données disponibles.

Dans l'espace des représentations, pondérer l'algorithme des plus proches voisins (par exemple par l'inverse de la distance, ou par la fréquence du rôle du voisin) peut permettre de capturer davantage d'information. On pourrait surpondérer les rôles sémantiques sous-représentés, ou difficiles à détecter, notamment.

Il serait également intéressant d'utiliser d'autres représentations distribuées de mots (entrainées par d'autres systèmes que SENNA), pour comparer les résultats. Des expériences préliminaires avec les représentations distribuées par Turian *et al.* (2010) [9] montrent des résultats semblables et encourageants. Les clusters lexicaux de Brown pourraient être employés avec la même idée.

Enfin, notre modèle prend actuellement toutes ses décisions de manière indépendante. Les arguments d'un même cadre (dans la même phrase) gagneraient à être étiquetés conjointement. Das *et al.* (2010) explorent cette idée, et gagne en précision, contre une petite perte de rappel à cause des contraintes supplémentaires. Les gains sont limités, notament à cause du faible nombre d'arguments à annoter par cadre (moins de 2 en moyenne). Nous avons mené quelques expériences dans cette direction, mais les résultats ne sont pas concluants.

7 Conclusion

FrameNet définit un ensemble de cadres sémantiques appelés par des prédicats, ainsi que les rôles pouvant être remplis par les arguments du dit prédicat. Nous avons employé des représentations distribuées de mots, entrainées par SENNA, pour améliorer la tâche de classification des arguments en rôles, en supposant connus le prédicat, le cadre sémantique, et la position des arguments.

Les représentations de mots situent les mots du lexique, *via* leurs coordonnées, dans un espace, l'espace des représentations. La représentation des arguments à classifier par leur mot le plus représentatif – dans nos expériences, leur tête syntaxique – permet de les situer eux-mêmes dans l'espace des représentations. Dès lors, on peut utiliser l'algorithme des plus proches voisins, ou bien partitionner l'espace suivant le rôle *en moyenne* le plus proche, pour classifier les arguments, et faire des prédictions raisonnables, un peu inférieures à l'état de l'art.

En utilisant ces prédictions dans le cadre d'un modèle à maximum d'entropie utilisant des caractéristiques descriptives de l'argument dans la phrase (notre système de référence), on obtient un modèle performant, légèrement supérieur à SEMAFOR, système à l'état de l'art, sur la tâche évaluée. En particulier, on remarque que les cadres sémantiques avec le moins d'exemples d'entrainement profitent davantage de la généralisation apportée par les représentations de mots, par rapport au modèle de référence.

Références

BROWN P. F., DESOUZA P. V., MERCER R. L., PIETRA V. J. D. & LAI J. C. (1992). Class-based n-gram models of natural language. *Comput. Linguist.*, **18**(4), 467–479.

COLLOBERT R., WESTON J., BOTTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, **12**, 2493–2537.

DAS D., SCHNEIDER N., CHEN D. & SMITH N. A. (2010). Probabilistic frame-semantic parsing. In *Human Language Technologies : The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, p. 948–956, Stroudsburg, PA, USA : Association for Computational Linguistics.

DAS D. & SMITH N. A. (2011). Semi-supervised frame-semantic parsing for unknown predicates. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies - Volume 1*, HLT '11, p. 1435–1444, Stroudsburg, PA, USA : Association for Computational Linguistics.

DE MARNEFFE M.-C., MACCARTNEY B. & MANNING C. D. (2006). Generating typed dependency parses from phrase structure parses. In *IN PROC. INT'L CONF. ON LANGUAGE RESOURCES AND EVALUATION (LREC*, p. 449–454.

9. http://metaoptimize.com/projects/wordreprs/

FILLMORE C. J. (1976). Frame semantics and the nature of language. *Annals of the New York Academy of Sciences*, **280**(1), 20–32.

FLEISCHMAN M. & HOVY E. (2003). A maximum entropy approach to framenet tagging. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology : companion volume of the Proceedings of HLT-NAACL 2003–short papers - Volume 2*, NAACL-Short '03, p. 22–24, Stroudsburg, PA, USA : Association for Computational Linguistics.

FLEISCHMAN M., KWON N. & HOVY E. (2003). Maximum entropy models for framenet classification. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, EMNLP '03, p. 49–56, Stroudsburg, PA, USA : Association for Computational Linguistics.

GILDEA D. & JURAFSKY D. (2000). Automatic labeling of semantic roles. In *ACL* : ACL.

JOHANSSON R., HEPPIN K. F. & KOKKINAKIS D. (2012). Semantic role labeling with the swedish framenet. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, M. U. DOĞAN, B. MAEGAARD, J. MARIANI, A. MORENO, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey : European Language Resources Association (ELRA).

JOHANSSON R. & NUGUES P. (2007). Lth : semantic structure extraction using nonprojective dependency trees. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, p. 227–230, Stroudsburg, PA, USA : Association for Computational Linguistics.

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, p. 768–774, Stroudsburg, PA, USA : Association for Computational Linguistics.

PUNYAKANOK V., ROTH D. & YIH W. (2008). The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, **34**(2).

SURDEANU M., HARABAGIU S., WILLIAMS J. & AARSETH P. (2003). Using predicate-argument structures for information extraction. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, p. 8–15, Stroudsburg, PA, USA : Association for Computational Linguistics.

TURIAN J., RATINOV L. & BENGIO Y. (2010). Word representations : a simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, p. 384–394, Stroudsburg, PA, USA : Association for Computational Linguistics.

Cross-lingual Word Sense Disambiguation for Predicate Labelling of French

Lonneke van der Plas[1] Marianna Apidianaki[2]
(1) IMS, Pfaffenwaldring 5B, 70569 Stuttgart, Germany
(2) LIMSI-CNRS, Rue John von Neumann, Campus Universitaire d'Orsay
Bât 508, 91405 Orsay Cedex, France
Lonneke.vanderPlas@ims.uni-stuttgart.de, Marianna.Apidianaki@limsi.fr

Résumé. Nous abordons la question du transfert d'annotations sémantiques, et plus spécifiquement d'étiquettes sur les prédicats, d'une langue à l'autre sur la base de corpus parallèles. Des travaux antérieurs ont transféré ces annotations directement au niveau des tokens, conduisant à un faible rappel. Nous présentons une approche globale de transfert qui agrège des informations repérées dans l'ensemble du corpus parallèle. Nous montrons que la performance de la méthode globale est supérieure aux résultats antérieurs en termes de rappel sans trop affecter la précision.

Abstract. We address the problem of transferring semantic annotations, more specifically predicate labellings, from one language to another using parallel corpora. Previous work has transferred these annotations directly at the token level, leading to low recall. We present a global approach to annotation transfer that aggregates information across the whole parallel corpus. We show that this global method outperforms previous results in terms of recall without sacrificing precision too much.

Mots-clés : transfert inter-langue, annotation sémantique automatique, prédicats, désambiguïsation lexicale, corpus parallèles.

Keywords: cross-lingual transfer, automatic semantic annotation, predicates, Word Sense Disambiguation, parallel corpora.

1 Introduction

There has recently been a large interest in multilingual natural language processing. Several annotation efforts have been devoted to developing resources for different languages, needed for supervised learning (Hajič *et al.*, 2009). However, there is a large number of languages that still lack linguistically annotated resources. For example for French, one of the most important European languages, there exists no corpus with predicate-argument annotations.

Predicate-argument annotations are the representation of the grammatically relevant aspects of a sentence meaning. This level of analysis provides a means of expressing a relation between syntactically different sentences, such as the sentence with the transitive verb in (1a) and the one with the intransitive verb in (1b). The semantic label *theme* expresses the fact that the object in (1a) has the same conceptual relation with the verb as the subject in (1b).

(1) a. [AGENT Mary] [REL-STOP.01 stopped] [THEME the car].
b. [THEME The car] [REL-STOP.01 stopped].

Semantic parsing or semantic role labelling refers to the task of automatically labelling predicates and arguments with predicate-argument structure. This task can be divided in two parts. One part refers to the labelling of the predicates with a predicate sense, and the other to the labelling of the arguments with semantic roles. Semantic parsers and the results of cross-lingual annotation transfer are therefore evaluated on both tasks : predicate labelling and role labelling. We focus on predicate labelling in this paper.

Cross-linguistically, the predicate-argument structure of a sentence is considered to be more stable than its syntactic form. The English sentence in (2a) can be considered as equivalent to the French sentence in (2b), despite the fact that the

position of their syntactic subject is occupied by different kinds of lexical elements and that the complements of the verbs differ both syntactically and semantically.

(2) a. [EXPERIENCER Mary] [REL-LIKE.01 liked] [CONTENT the idea]. (English)
b. [CONTENT L'idée] a [REL-LIKE.01 plu] [EXPERIENCER à Marie]. (French)

Since manual annotation is a costly and time-consuming approach to resource development, cross-lingual annotation transfer offers an attractive alternative. In this approach, predicate-argument annotations on a source language, for which there exists semantic annotations, are transferred to a target language using parallel corpora (Padó, 2007; Basili *et al.*, 2009; Annesi & Basili, 2010; van der Plas *et al.*, 2011). The increased stability of predicate-argument structures across languages makes their cross-lingual transfer attractive when compared with, for example, syntactic structures.

Traditional methods for cross-lingual transfer rely on the semantic equivalence of the original and the translated sentences, and on correct and complete alignments between words or constituents in those sentences. Since the semantic annotations are transferred directly from token to token, we will refer to these traditional methods as direct cross-lingual transfer. Alignment errors and translation shifts represent major sources of mistakes in the direct transfer approach which result in incorrect and incomplete annotations in the target language.

In this paper, we propose a different strategy to predicate labelling that is less sensitive to alignment errors and translation shifts. Instead of transferring semantic annotations on a token-to-token basis, we aggregate information across the whole parallel corpus to correct token-level mistakes resulting from direct cross-lingual transfer. Our approach consists of two steps : In the learning step, a global model is learned on the basis of source language (English) predicate annotations in a word-aligned English-French training corpus. In the labelling step, this model assigns predicate labels to verbs in target language texts (here, French). We model cross-lingual transfer of predicate labels as a cross-lingual word sense disambiguation (WSD) task, because this fits well with the lexical nature of the task : annotating French verbs with English predicate labels.

Our contributions are two-fold. First, we present a global approach to semantic annotation transfer that corrects token-level mistakes as found in traditional direct transfer methods. Second, we show the strengths and limitations of global vs direct transfer.

In the next section, we present related work on cross-lingual annotation transfer. We then briefly discuss the semantic annotation framework we are using and explain why we decided to use an English semantic framework for annotating French. In Section 4, we briefly present the direct transfer method. In Section 5, we describe the tools and data we use and explain the adopted evaluation framework. In Section 6, we explain the global method proposed in this paper. The results are presented in Section 7, before concluding.

2 Related work

Transferring annotations from one language to another in order to train monolingual tools for new languages was first introduced by Yarowsky & Ngai (2001). In their approach, token-level part-of-speech (PoS) and noun phrase bracketing information was projected across word-aligned bitext and this partial annotation served to estimate the parameters of a model that generalized from the noisy projection in a robust way. In more recent work, Das & Petrov (2011) propose a graph-based framework for projecting syntactic information across language boundaries. They create type-level tag dictionaries by aggregating over projected token-level information extracted from bi-text and use label propagation on a similarity graph to smooth and expand the label distributions. A different approach to cross-lingual PoS tagging is proposed by Täckström *et al.* (2013) who couple token and type constraints in order to guide learning. These two types of information are viewed as complementary : token-level projections offer precise constraints for tagging in a particular context while broad coverage type-level dictionaries help to filter noise in token-level projections. Our approach to cross-lingual predicate labelling follows this vein. Instead of solely relying on token-level information acquired from word-alignments, we combine this with type-level information captured by our global method which is trained on the entire corpus. We, however, are concerned with semantic annotations and not PoS tags.

Transfer of semantic annotation has started off with direct transfer of FrameNet semantic annotations (Padó, 2007; Basili *et al.*, 2009; Annesi & Basili, 2010). With the addition of a learning step and the use of PropBank data, Van der Plas et al. (2011) have scaled up previous efforts. They show that a joint semantic-syntactic parser trained on the output of direct transfer and additional syntactic annotations produces better parses than the input it received by aggregating information

Frame	Semantic roles
pay.01	A0 : payer or buyer A1 : money or attention A2 : person being paid, destination of attention A3 : commodity, paid for what
pay.02 *pay off*	A0 : payer A1 : debt A2 : owed to whom, person paid
pay.03 *pay out*	A0 : payer or buyer A1 : money or attention A2 : person being paid, destination of attention A3 : commodity, paid for what
pay.04	A1 : thing succeeding or working out
pay.05 *pay off*	A1 : thing succeeding or working out
pay.06 *pay down*	A0 : payer A1 : debt

TABLE 1 – The PropBank lexicon entry for *pay*.

across multiple examples. Our method is more resource-light, as we do not need syntactic annotations neither on the target nor on the source side.

The same emphasis on learning is found in cross-lingual model transfer, where source language models are adapted to work on the target language directly. For semantic role labelling, Kozhevnikov & Titov (2013) use shared feature representations (syntactic and lexical) to adapt a source model to a target-language model. They, however, do not consider the task of predicate labelling but only semantic role labelling.

In this work, we address predicate labelling in languages other than English as a cross-lingual WSD task. Word sense disambiguation is the task of automatically identifying the meaning of words in context (Navigli, 2009). In its cross-lingual variant, the candidate senses are the words' translations in other languages and WSD aims at predicting semantically correct translations for instances of the words in context (Resnik & Yarowsky, 2000; Ng *et al.*, 2003; Carpuat & Wu, 2007; Apidianaki, 2009). In our experiments, we apply the cross-lingual WSD method employed by Apidianaki *et al.* (2012) for improving the quality of Machine Translation, in a different setting : instead of assigning semantically appropriate translations to words in context, the WSD classifier serves for selecting the most adequate English predicate label for verbs in a new language. More details on the adaptation of the method to predicate labelling are given below, in Section 6 of the paper.

3 The semantic annotation framework

There exist three frameworks for annotating corpora with predicate-argument structure : FrameNet (Fillmore *et al.*, 2003), VerbNet (Kipper, 2005) and PropBank (Palmer *et al.*, 2005). We chose PropBank for applying our predicate labelling method. In the following, we describe PropBank in more detail as well as the criteria that led to choosing this framework as well as how the English PropBank can be used to annotate French.

3.1 The Proposition Bank

The Proposition Bank (PropBank) is a linguistic resource that contains information on the semantic structure of sentences (Palmer *et al.*, 2005). It consists of a one-million-word corpus of naturally occurring sentences annotated with semantic structures and a lexicon (the PropBank frame files) that lists all the predicates (verbs) that can be found in the annotated sentences and the sets of semantic roles they introduce.

Predicates are marked with labels that specify the sense of a verb in a particular context. Each lemma described in the frame files (3300 verbs) contains one or more lexemes (4500 verb senses), which are used as predicate labels. The PropBank frame files specify the interpretation of the roles for each verb in its different senses. The interpretation of the numbered roles is given for each lexeme separately. Table 1 illustrates the entry for the verb *pay* in the PropBank frame files.

The semantic role annotation is based on Dowty's theory of Proto-Roles (Dowty, 1991). Arguments are marked with the labels A0 to A5, which represent semantic roles of a very general kind. Only the labels A0 and A1 have approximately the same value with all verbs : they are used to mark instances of proto-agent (A0) and proto-patient (A1). The meaning of other numbered arguments is verb-dependent. It depends on the meaning of the verb, on the type of the constituent they are assigned to, and on the number of roles present in a particular sentence. A3, for example, can mark purpose as is the case in (3), or it can mark direction or some other role with other verbs. The indices are assigned according to the roles' prominence in the sentence. More prominent are the roles that are more closely related to the verb. The AM-* labels can be specified further as : location, cause, extent, time, discourse connectives, purpose, general purpose, manner, direction. The labels for adjuncts are more specific but less verb-specific than the labels for arguments, and they do not depend on the presence of other roles in the sentence.

(3) [$_{A0}$ The Latin American nation] has [$_{REL-PAY.01}$ paid] [$_{A1}$ very little] [$_{A3}$ on its debt] [$_{AM-TMP}$ since early last year].

Although PropBank is considered as the most language-specific of the three resources, Samardžić *et al.* (2010) motivate the use of PropBank for annotation transfer with two reasons. First, the lexicon in this resource is corpus-driven. It is built by extracting and describing all the predicates that occur in a predefined sample of naturally occurring sentences. Since the aim is to annotate a corpus of naturally occurring sentences exhaustively, we expect that such a lexicon can provide a better coverage than the lexicon in FrameNet and VerbNet, which are not corpus-driven. Second, the labels used in PropBank both for predicates and arguments involve fewer theoretical assumptions than the labels in FrameNet. While the FrameNet labels capture mostly linguistic intuition at the targeted level of lexical semantics and the relations between the lexical items, the PropBank labels rely strongly on the observable behaviour of words. The distinction between the different verb senses, for instance, is made taking into account the different sets of arguments and other observable differences, such as the presence of the particle that distinguishes pay.04 from pay.05 in Table 1. This approach can be expected to provide more tangible criteria for annotators in deciding how to annotate each instance of the predicate-argument structure found in the corpus, ensuring a more reliable and more consistent annotation. Also, it enables a more direct comparison of the structures across languages, since the representation of the structures does not include any hypothesized levels of abstraction.

3.2 Using the English PropBank to annotate French

Adapting a semantic framework to a new language is a time-consuming process. In order to generate broad-coverage annotations for a target language in limited time, we transfer semantic annotations from the source language directly to the target language without adapting the semantic annotation framework to the target language. This means that French verbs will be annotated with English predicate senses. For this to work, we need to show that PropBank is cross-lingually valid. Van der Plas et al. (2010) did this in a manual annotation effort. For a complete description of the annotation procedure, that involved four evaluators and several stages, we point the reader to Van der Plas et al. (2010). In this section, we summarize the procedure and briefly discuss the main outcomes.

In this manual annotation study, annotators used the English PropBank frame files to annotate French sentences. This means that for every predicate found in a French sentence, they needed to translate it and find an English verb sense that was applicable to the French verb. If an appropriate entry could not be found in the frame files for a given predicate, the annotators were instructed to use the dummy label for the predicate and fill in the roles according to their own insights.

Some of the differences in annotation observed between annotators were due to lexical variation in English. For example, if one annotator put the label 'demonstrate.01' whereas the other used 'show.01' on an instance of the verb *montrer*, this should not be counted as a disagreement as the two senses are linked. Therefore, agreement scores were provided both on the basis of the predicate sense labels and on the basis of the verb class, using the verb classifications from VerbNet (Kipper, 2005) and the mapping to PropBank labels as given in the type mappings of the SemLink project [1] (Loper *et al.*, 2007). VerbNet is a hierarchically organised verb lexicon of English verbs (Kipper, 2005). It is organized into verb classes extending Levin (1993) classes through refinement and addition of subclasses to achieve syntactic and semantic coherence among members of a class. The senses 'demonstrate.01' and 'show.01' appear in the same verb class according to the type mappings of the SemLink project and were thus considered as correct.

The average inter-annotator agreement reported in these experiments was relatively low when the annotations on the PropBank verb sense level were compared : 59%. However, at the level of verb classes, the inter-annotator agreement

1. http://verbs.colorado.edu/semlink/

increased to 81%. The authors identify collocations and idiomatic expressions as the main sources of disagreement in predicate labellings among annotators, as is also shown in studies on other language pairs (Burchardt *et al.*, 2009).

For a single annotator, the main measure of cross-lingual validity was the percentage of dummy predicates in the annotation. Van der Plas (2010) found 130 dummy annotations in 1000 sentences. A manual classification of the dummy labels showed that the dummy label was mainly used for French multi-word expressions (82%), most of which could be translated by a single English verb (47%) whereas others could not because they were translated by a combination that included a form of 'be' that was not annotated in PropBank (25%). The 47% of multi-word expressions that received the dummy label showed the annotator's reluctance to put a single verb label on a French multi-word expression. The annotation guidelines could however be adapted to instruct annotators not to hesitate in such cases. Based on these findings, the authors conclude that the annotation framework PropBank is cross-lingually valid.

4 Direct cross-lingual transfer

Before presenting our global method for predicate labelling, we would like to remind the reader of the method for direct cross-lingual transfer which is used in comparisons and combinations throughout this paper. It is taken from Van der Plas et al. (2011), but we give a short summary here for the readers convenience. The method is based on the Direct Correspondence Assumption for syntactic dependency trees by Hwa *et al.* (2005).

> **Direct Semantic Transfer (DST)** For any pair of sentences E and F that are translations of each other, we transfer the semantic relationship $R(x_E, y_E)$ to $R(x_F, y_F)$ if and only if there exists a word-alignment between x_E and x_F and between y_E and y_F, and we transfer the semantic property $P(x_E)$ to $P(x_F)$ if and only if there exists a word-alignment between x_E and x_F.

The properties that are transferred through DST are predicate senses. The relationships that are transferred are semantic role dependencies, but we are not concerned with them in this paper. The properties are transferred from the English side of a parallel corpus that is automatically annotated with syntactic-semantic analyses to the foreign language side, as described in the following section.

5 Tools and data

The parallel corpus used in our experiments is the English-French part of the Europarl corpus (Koehn, 2005). The English part of the parallel corpus is annotated by a freely-available syntactic-semantic parser (Henderson *et al.*, 2008; Titov *et al.*, 2009) trained on the CoNLL 2009 training set (the Penn Treebank corpus (Marcus *et al.*, 1993) merged with PropBank labels (Palmer *et al.*, 2005) and NomBank labels [2] (Meyers, 2007)). In the experiments presented in this paper, we only use the predicate labels found in the English part of Europarl, omitting the labels assigned to the arguments.

For the direct transfer of semantic annotations from the English to the French side of the parallel corpus, we use the method described in Section 4. As is usual practice in pre-processing for automatic word alignment, both parts of the parallel corpus were tokenised and lowercased and only sentence pairs corresponding to a one-to-one sentence alignment, with lengths ranging from one to 40 tokens on both French and English sides, were considered. We subsequently word aligned the English and French sentences automatically using GIZA++ (Och & Ney, 2003) in both translation directions and retained only intersecting alignments. Furthermore, because translation shifts are known to pose problems for the automatic projection of semantic annotation across languages (Padó, 2007), we selected only those parallel sentences in Europarl that are direct translations from English to French, or vice versa. In the end, we obtained a word-aligned parallel corpus of 276-thousand sentence pairs.

For testing, we use the hand-annotated data described in Van der Plas et al. (2010). We randomly split those 1000 sentences into test and development set containing 500 sentences each. We use the development set for the current experiments, which contains 879 predicates.

2. We limit our experiments to verbal predicates only because the semantic annotations on French test sentences are limited to verbal predicates. Even though verbal predicates in the target language can be expressed as non-verbal predicates in the source language, the transfer of Nombank labels to verbal predicates is not straightforward due to difficulties in mapping between the two annotation frameworks.

6 Global cross-lingual predicate labelling

Traditional cross-lingual transfer methods are locally defined. Transfer takes place on a token-to-token basis and, as a consequence, missing or incorrect alignments lead immediately to missing and incorrect annotations in the target language. Our method for cross-lingual predicate labelling is globally defined and relies less on actual alignments.

Our aim is to put predicate labels that originate from the English side of the parallel corpus on the French verbs in the other side of the corpus. The predicate labels contain the English verb and its sense. For example, "give.01" stands for the first sense of the verb *give*. As the predicate label contains a lot of lexical information, assigning the correct English predicate label to a French verb is a task very close to word sense disambiguation (WSD), which aims at automatically identifying the meaning of words in context (Navigli, 2009). In cross-lingual WSD, the candidate senses of words are their translations in other languages from which the most adequate has to be selected for contextualized instances of the words (Carpuat & Wu, 2007; Apidianaki, 2009). The main difference between cross-lingual WSD and our cross-lingual transfer of predicate labels is that we do not search for correct translations of French words but for the most appropriate predicate labels in context (i.e. verbs disambiguated with a predicate sense).

The global predicate labelling method that we propose consists of a learning step and a labelling step. During learning, we compute estimates for annotation transfer on the basis of the word alignments between English and French predicates over the entire parallel corpus. At labelling time, we label French verbs with English predicate labels without need for parallel data or alignments. The method is language-independent and only requires minimal linguistic resources.

In contrast to direct transfer, we provide a predicate label for all French verbs in the test set, not only aligned ones. We expect to augment the recall when using global estimates and hope that the affect on precision is not too negative.

6.1 Pre-processing

The WSD classifier is trained on the Europarl corpus tagged with PropBank information on the English side, as described in Section 5. To identify the sets of candidate predicate labels for each French verb, we replace the English verbs by the corresponding predicate label wherever this is available. Then we tag both parts of the corpus by part of speech (PoS) using the TreeTagger (Schmid, 1994) and rebuild the parallel files (one sentence per line) by replacing words on both sides by the corresponding 'lemma_PoS tag' pair, and keeping the predicate labels in the place of English verbs. The corpus is then aligned at the word level in both directions using GIZA++ (Och & Ney, 2003) and a lexicon is built from intersecting alignments. Lexicon entries for French verbs contain the labels to which they were aligned in the training corpus. The entry for the verb *encourager*, for instance, contains seven predicate labels : {urge.01, foster.01, stimulate.01, promote.02, encourage.01, encourage.02, renew.01}, two of which correspond to the same English verb (encourage). We keep labels with a high alignment confidence score according to GIZA++ and experiment with two thresholds, 0.01 and 0.001. Naturally, the second threshold retains a higher number of candidate labels.

6.2 Learning

For each French verb (v) in the lexicon, we want to be able to identify its correct predicate label in a new context. A feature vector is built for each candidate label following the procedure described in Apidianaki *et al.* (2012). For each candidate label L_i of a French verb v, we extract the content word co-occurrences of v in the sentences where it translates an English verb tagged with the label L_i. The retained French words constitute the features of the vector built for the label. Let N be the number of features retained for each label L_i of v from the corresponding French contexts. Each feature F_j ($1 \leq j \leq N$) receives a total weight with the label $\text{tw}(F_j, L_i)$ learned from the data and defined as the product of the feature's global weight, $\text{gw}(F_j)$, and its local weight with that label, $\text{lw}(F_j, L_i)$. The global weight of a feature F_j is a function of the number NL_i of labels (L_i's) to which F_j is related, and of the probabilities (p_{ij}) that F_j co-occurs with instances of v corresponding to each of the L_i's :

$$\text{gw}(F_j) = 1 - \frac{\sum_{L_i} p_{ij} \log(p_{ij})}{NL_i} \quad (1)$$

Each p_{ij} is computed as the ratio of the co-occurrence count of F_j with v when it corresponds to a label L_i to the total number of features (N) seen with L_i in the corpus :

$$p_{ij} = \frac{\text{cooc\_count}(F_j, L_i)}{N} \tag{2}$$

The local weight $\text{lw}(F_j, L_i)$ between a feature F_j and a label L_i directly depends on their co-occurrence count :

$$\text{lw}(F_j, L_i) = \log(\text{cooc\_count}(F_j, L_i)) \tag{3}$$

The intuition underlying this weighting scheme is that if an interesting semantic relation exists between a feature F_j and a specific predicate label L_i of a verb v, then we expect the probability (p_{ij}) of the feature F_j occurring in the contexts where v is translated by this label to be larger than if they were independent. In other words, a feature gets a high total weight (tw) with a label when it appears frequently in the corresponding French contexts and rarely in the contexts of the other labels.

6.3 Labelling

Predicate identification is done by selecting verbs based on the PoS labels provided by the tagger and subsequently filtering out modals and instances of the verb *être*.[3] The most suitable predicate labels are then assigned to the retained French verbs by our disambiguation classifier. The weighted feature vectors built for the candidate labels of a French verb as described in the previous section are compared to the context of a new instance of the verb and an association score is assigned to each candidate label. To facilitate comparison with the vectors, the new contexts (sentences) are lemmatized and PoS tagged on the fly (with TreeTagger) and the content word co-occurrences of the French verb are gathered in a bag of words. If common features are found between the new context and the vector of a label, their association score corresponds to the mean of the weights of their shared features with that label (i.e. found in its vector). In Equation 4, $(CF_j)_{j=1}^{|CF|}$ is the set of common features between the label vector V_i and the new context C and tw is the weight of a CF with label L_i, computed as explained in the previous section.

$$\text{assoc\_score}(V_i, C) = \frac{\sum_{j=1}^{|CF|} \text{tw}(CF_j, L_i)}{|CF|} \tag{4}$$

The label that receives the highest association score with the new context is returned and serves to annotate the corresponding French verb. For example, among the candidate labels for the verb *encourager* ({urge.01, foster.01, stimulate.01, promote.02, encourage.01, encourage.02, renew.01}), the classifier selects the predicate label *encourage.02* for the following instance :

> D'ailleurs, le rapport von Wogau, que vient de voter le Parlement européen *encourage* [encourage.02] en ce sens.

The label selected in this case corrects the label [support.01] that was assigned through direct transfer.

7 Results and discussion

We run experiments using the global method for predicate labelling described in the previous section and compare the results to the ones obtained through direct transfer. The results are presented in Table 2 where they are also compared to upper bounds from manual annotations and previous work.

The first row of Table 2 shows the results from using the traditional direct transfer method. The second and third rows present the results obtained using the global method, where we use cross-lingual WSD to label predicates. In row 2, we present the results obtained when using an alignment confidence threshold of 0.01 (retaining labels with an alignment score above 0.01, according to GIZA++) and in row 3, the results obtained using a threshold of 0.001. For comparison, we show results when using a parser as in Van der Plas et al. (2011) who use a joint syntactic-semantic parser and

3. We exclude the verb *être* because its English counterpart (*be*) is not annotated in the CoNLL-2009 data used in our experiments.

	Predicate senses			Verb classes		
	P	R	F	P	R	F
Direct Transfer	51	29	37	75	38	50
CLWSD(0.01)	45	39	42	73	61	67
CLWSD(0.001)	42	40	41	70	65	67
Parser	56	46	51	83	63	72
Manual	61	57	59	85	76	81

TABLE 2 – Percent recall, precision and F-measure for predicate labelling.

syntactic annotations on French to do predicate and semantic role labelling. [4] We show an upper bound in the last row. This represents the inter-annotator agreement for manual annotation on a random set of 100 sentences taken from data provided by Van der Plas et al. (2010). Two evaluation settings were used in that work, in order to avoid penalizing synonymous verb senses assigned by the annotators : the inter-annotator agreement reached at the verb sense label was compared to the agreement reached using verb classes, as explained in Section 3.2. Because we do not want to penalize the predicate labelling system for selecting verb senses that are synonyms of the verb senses in the gold, we follow the same strategy and take verb classes into account during evaluation. More precisely, we calculate scores based on the exact correspondence between the label proposed by our system and the gold label found in the test data, and we also perform a coarser evaluation taking verb classes into account. The first evaluation is too strict as it penalizes the system when it selects a predicate sense that is synonymous to the gold sense or a predicate that belongs to the same word class. The results headed by verb classes evaluate on a more realistic basis, capturing semantic correspondences beyond surface variations. In this setting, predicate labels are correct if they belong to the same verb class as the predicate in the gold annotations.

When we look at the differences between the three automatic methods for the evaluation on predicate senses, we see that for the direct transfer method especially recall is very low, 29%. The global method (alignment score threshold = 0.01) has a much better recall, 39%. Precision is lower but the F-score increases by 5 percentage points. When the alignment confidence threshold is lowered to 0.001, which means that more candidate labels are retained, recall increases and precision goes down, as expected. As explained above, the parser from Van der Plas et al. (2011) (shown in row 4) has access to both PoS and syntactic information on the target side and uses a joint syntactic-semantic framework. When we take its dependence on two extra resources into consideration, its performance is not that impressive. However, we can learn from these results that structural information is beneficial. Nevertheless, our results show that the cross-lingual WSD method which relies on much less external knowledge, outperforms the direct method on both senses and verb classes. In future work, we plan to include word position information in our cross-lingual WSD method. This will give the method access to structural information while staying knowledge-lean.

When we look at the results using verb classes which permit to abstract from surface variations and capture semantic correspondences, the overall performance numbers are higher as expected. More importantly, the difference in performance between the cross-lingual WSD method using the more restrained set of labels (alignment score threshold = 0.01), which performs best, and the direct transfer method are now much larger (three times as important, from 5 to 17 percentage points) whereas the difference between our method and the parser is further reduced (from 10 to 5 percentage points). The differences between the parser and our method can be mainly attributed to arbitrary variations between predicate labels that belong to the same verb class. When we look at the precision and recall scores we see that the cross-lingual WSD method with the threshold set at '0.01' improves the recall of the direct transfer method by 23 percentage points, whereas precision only drops by 2 points. The cross-lingual WSD method that needs only a PoS tagger on the target and no syntactic annotation nor parsing frameworks, results in much better scores than direct transfer that is equally knowledge-light. All automatic methods are still quite far from the results from manual annotation.

In summary, these results show that the global cross-lingual WSD method for predicate labelling improves recall of direct transfer methods without sacrificing precision too much. In future work, we plan to combine the direct transfer method and the global cross-lingual WSD method, because the two are complementary in terms of recall and precision.

An example of predicate labelling might help the reader to get an idea of the contribution of the global method. In Table 3, we present an example where cross-lingual WSD annotates more verbs than the direct transfer : labels [stress.01] and [seem.01], assigned during disambiguation, are missing from the first sentence after transfer. Moreover, it would be impossible to get these labels through direct transfer from the English source sentence because they are simply not there,

4. The results are different from the results reported in Van der Plas et al. (2011) because we used the development set in our evaluations.

English (automatic) : There is in particular one amendment, let [let.01] me point [point.02] out, concerning [concern.01] the energy sector, which, in my capacity as rapporteur, I see [see.01] as particularly important.
Transfer : Il y a notamment un amendement, je le souligne, concernant [concern.01] le secteur de l'énergie, qui me paraît en tant que rapporteur particulièrement important.
CLWSD : Il y a notamment un amendement, je le souligne [stress.01], concernant [concern.01] le secteur de l'énergie, qui me paraît [seem.01] en tant que rapporteur particulièrement important.

TABLE 3 – Predicate label addition and correction using CLWSD.

due to the non-literal translation. This example shows the limitations of token-to token, direct transfer and how the global method is able to compensate for that by using information aggregated across the whole parallel corpus.

8 Conclusion

In this paper, we present a knowledge-light global approach to the cross-lingual transfer of semantic annnotation that aggregates information across the whole parallel corpus. Previous work has transferred annotations directly from token to token in parallel sentences leading to low recall and token-level mistakes. We show how the global method, based on cross-lingual word sense disambiguation, improves recall by a large margin without sacrificing precision too much.

Given the knowledge-lean character of the proposed method, in future work we plan to apply it for cross-lingual predicate labelling in other language pairs. Furthermore, we would like to include target side structural information (e.g. word position information) in the cross-lingual WSD method. Last but not least, we intend to work towards global methods for role identification and labelling which will allow to propose a complete SRL annotation framework based on global information. In that respect, we would also like to try and combine global and direct methods because the two seem complementary in terms of recall and precision.

Références

ANNESI P. & BASILI R. (2010). Cross-lingual alignment of FrameNet annotations through Hidden Markov Models. In *Proceedings of CICLing*.

APIDIANAKI M. (2009). Data-driven Semantic Analysis for Multilingual WSD and Lexical Selection in Translation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL-09)*, p. 77–85, Athens, Greece.

APIDIANAKI M., WISNIEWSKI G., SOKOLOV A., MAX A. & YVON F. (2012). WSD for n-best reranking and local language modeling in SMT. In *Proceedings of the Sixth Workshop on Syntax, Semantics and Structure in Statistical Translation*, p. 1–9, Jeju, Republic of Korea : Association for Computational Linguistics.

BASILI R., CAO D. D., CROCE D., COPPOLA B. & MOSCHITTI A. (2009). *Computational Linguistics and Intelligent Text Processing*, chapter Cross-Language Frame Semantics Transfer in Bilingual Corpora, p. 332–345. Springer Berlin / Heidelberg.

BURCHARDT A., ERK K., FRANK A., KOWALSKI A., PADO S. & PINKAL M. (2009). *Multilingual FrameNets in Computational Lexicography : Methods and Applications*, chapter FrameNet for the semantic analysis of German : Annotation, representation and automation, p. 209–244. De Gruyter Mouton, Berlin.

CARPUAT M. & WU D. (2007). Improving Statistical Machine Translation using Word Sense Disambiguation. In *Proceedings of the Joint EMNLP-CoNLL Conference*, p. 61–72, Prague, Czech Republic.

DAS D. & PETROV S. (2011). Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies*, p. 600–609, Portland, Oregon, USA : Association for Computational Linguistics.

DOWTY D. (1991). Thematic proto-roles and argument selection. *Language*, **67.3**, 547–619.

FILLMORE C. J., JOHNSON R. & PETRUCK M. (2003). Background to FrameNet. *International journal of lexicography*, **16.3**, 235–250.

HAJIČ J., CIARAMITA M., JOHANSSON R., KAWAHARA D., MARTÍ M. A., MÀRQUEZ L., MEYERS A., NIVRE J., PADÓ S., ŠTEPÁNEK J., STRAŇÁK P., SURDEANU M., XUE N. & ZHANG Y. (2009). The CoNLL-2009 shared

task : Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009)*.

HENDERSON J., MERLO P., MUSILLO G. & TITOV I. (2008). A latent variable model of synchronous parsing for syntactic and semantic dependencies. In *Proceedings of CONLL 2008*, p. 178–182.

HWA R., RESNIK P., A.WEINBERG, CABEZAS C. & KOLAK O. (2005). Bootstrapping parsers via syntactic projection accross parallel texts. *Natural language engineering*, **11**, 311–325.

KIPPER K. (2005). *VerbNet : A broad-coverage, comprehensive verb lexicon*. PhD thesis, University of Pennsylvania.

KOEHN P. (2005). Europarl : A Parallel Corpus for Statistical Machine Translation. In *Proceedings of MT Summit X*, p. 79–86, Phuket, Thailand.

KOZHEVNIKOV M. & TITOV I. (2013). Crosslingual transfer of semantic role models. In *In Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics*, Sofia, Bulgaria : Association for Computational Linguistics.

LEVIN B. (1993). *English Verb Classes and Alternations : A preliminary investigation*. Rapport interne, University of Chicago Press.

LOPER E., YI S.-T. & PALMER M. (2007). Combining lexical resources : Mapping between PropBank and VerbNet. In *Proceedings of the 7th International Workshop on Computational Semantics (IWCS-7)*, p. 118–129, Tilburg, The Netherlands.

MARCUS M., SANTORINI B. & MARCINKIEWICZ M. (1993). Building a large annotated corpus of English : the Penn Treebank. *Comp. Ling.*, **19**, 313–330.

MEYERS A. (2007). *Annotation guidelines for NomBank - noun argument structure for PropBank*. Rapport interne, New York University.

NAVIGLI R. (2009). Word Sense Disambiguation : a Survey. *ACM Computing Surveys*, **41**(2), 1–69.

NG H. T., WANG B. & CHAN Y. S. (2003). Exploiting Parallel Texts for Word Sense Disambiguation : An Empirical Study. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, p. 455–462, Sapporo, Japan.

OCH F. J. & NEY H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, **29**, 19–51.

PADÓ S. (2007). *Cross-lingual Annotation Projection Models for Role-Semantic Information*. PhD thesis, Saarland University.

PALMER M., GILDEA D. & KINGSBURY P. (2005). The Proposition Bank : An annotated corpus of semantic roles. *Computational Linguistics*, **31**, 71–105.

RESNIK P. & YAROWSKY D. (2000). Distinguishing Systems and Distinguishing Senses : New Evaluation Methods for Word Sense Disambiguation. *Natural Language Engineering*, **5**(3), 113–133.

SAMARDŽIĆ T., VAN DER PLAS L., KASHAEVA G. & MERLO P. (2010). The scope and the sources of variation in verbal predicates in English and French. In *Proceedings of the 9th International Workshop on Treebanks and Linguistic Theories*.

SCHMID H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, p. 44–49, Manchester, UK. http ://www.ims.uni-stuttgart.de/~schmid/.

TÄCKSTRÖM O., DAS D., PETROV S., MCDONALD R. & NIVRE J. (2013). Token and type constraints for cross-lingual part-of-speech tagging. In *Transactions of the ACL* : Association for Computational Linguistics.

TITOV I., HENDERSON J., MERLO P. & MUSILLO G. (2009). Online graph planarisation for synchronous parsing of semantic and syntactic dependencies. In *Proceedings of the twenty-first international joint conference on artificial intelligence (IJCAI-09)*, Pasadena, California.

VAN DER PLAS L., MERLO P. & HENDERSON J. (2011). Scaling up cross-lingual semantic annotation transfer. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics and the Human Language Technologies conference*.

VAN DER PLAS L., SAMARDŽIĆ T. & MERLO P. (2010). Cross-lingual validity of PropBank in the manual annotation of French. In *In Proceedings of the 4th Linguistic Annotation Workshop (The LAW IV)*, Uppsala, Sweden.

YAROWSKY D. & NGAI G. (2001). Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, NAACL '01, p. 1–8, Stroudsburg, PA, USA : Association for Computational Linguistics.

Améliorer l'étiquetage de "que" par les descripteurs ciblés et les règles

Assaf Urieli[1, 2]
(1) CLLE-ERSS: CNRS & Université de Toulouse, Toulouse, France
(2) Joliciel Informatique SARL, 2 avenue du Cardié, 09000 Foix, France
assaf.urieli@univ-tlse2.fr

Résumé. Les outils TAL statistiques robustes, et en particulier les étiqueteurs morphosyntaxiques, utilisent souvent des descripteurs "pauvres", qui peuvent être appliqués facilement à n'importe quelle langue, mais qui ne regarde pas plus loin que 1 ou 2 tokens à droite et à gauche et ne prennent pas en compte des classes d'équivalence syntaxiques. Bien que l'étiquetage morphosyntaxique atteint des niveaux élevés d'exactitude (autour de 97 %), les 3 % d'erreurs qui subsistent induisent systématiquement une baisse de 3 % dans l'exactitude du parseur. Parmi les phénomènes les plus faciles à cibler à l'aide de l'injection de connaissances linguistiques plus riches sont les mots fonctionnels ambigus, tels que le mot "que" en français. Dans cette étude, nous cherchons à améliorer l'étiquetage morphosyntaxique de "que" par l'utilisation de descripteurs ciblés et riches lors de l'entraînement, et par l'utilisation de règles symboliques qui contournent le modèle statistique lors de l'analyse. Nous atteignons une réduction du taux d'erreur de 45 % par les descripteurs riches, et de 55 % si on ajoute des règles.

Abstract. Robust statistical NLP tools, and in particular pos-taggers, often use knowledge-poor features, which are easily applicable to any language but do not look beyond 1 or 2 tokens to the right and left and do not make use of syntactic equivalence classes. Although pos-tagging tends to get high accuracy scores (around 97%), the remaining 3% errors systematically result in a 3% loss in parsing accuracy. Some of the easiest phenomena to target via the injection of richer linguistic knowledge are ambiguous function words, such as "que" in French. In this study, we attempt to improve the pos-tagging of "que" through the use of targeted knowledge-rich features during training, and symbolic rules which override the statistical model during analysis. We reduce the error rate by 45% using targeted knowledge-rich features, and 55% if we add rules.

Mots-clés : étiquetage morphosyntaxique, apprentissage automatique supervisé, descripteurs riches, systèmes statistiques robustes.

Keywords: pos-tagging, supervised machine learning, knowledge-rich features, robust statistical systems.

1 Introduction

Les outils TAL statistiques robustes sont relativement faciles à construire : il suffit de disposer d'un corpus d'apprentissage annoté, d'un classifieur robuste (ex. SVM linéaire), d'un algorithme d'analyse (ex. le parsing par transitions pour l'analyse syntaxique) et de quelques descripteurs. La plupart de ces systèmes utilisent des descripteurs linguistiquement pauvres, limités aux bigrammes ou trigrammes des tokens ou des étiquettes morphosyntaxiques, à quelques informations de base tirées d'un lexique à large couverture, et, au niveau du parsing, à un examen superficiel de la tête ou du dépendant le plus à droite ou à gauche d'un token donné. Même les études qui parlent de descripteurs "riches" (Zhang & Nivre, 2011) se limitent à des descripteurs génériques, qui prennent en compte des informations de surface telles que la valence d'un token (nombre de dépendants) ou la distance entre deux tokens, mais ne cherchent pas à coder les phénomènes spécifiques d'une langue donnée. Cela présente l'avantage d'une application facile à beaucoup de langues, mais nous empêche d'injecter des connaissances linguistiques spécifiques, et limite donc les gains d'exactitude possibles. Notre but principal ici est de trouver des moyens d'améliorer les analyses des systèmes statistiques par l'introduction d'informations plus riches.

L'analyseur syntaxique Talismane [1] a été développé dans l'optique de permettre à l'utilisateur d'injecter le maximum d'informations linguistiques, dans un système qui reste statistique et robuste (Urieli, 2013). Il comprend quatre modules

1. http://redac.univ-tlse2.fr/applications/talismane.html

statistiques enchaînés : la segmentation en phrases, la segmentation en mots (tokenisation), l'étiquetage morphosyntaxique (pos-tagging) et l'analyse syntaxique en dépendances par transitions (parsing), dont l'algorithme de base est décrit dans Kübler *et al.* (2009). Nous nous sommes intéressés tout particulièrement à l'interaction entre les différents modules. Dans une étude précédente, nous avons exploré la propagation des ambiguïtés de l'étiqueteur morphosyntaxique vers le parseur, afin que ce dernier puisse les corriger (Urieli & Tanguy, 2013). Dans cette étude, nous cherchons plutôt à améliorer l'étiquetage morphosyntaxique en amont du parseur, par l'injection des connaissances linguistiques spécifiques pour certains phénomènes particulièrement importants pour le parsing, se concentrant ici sur l'étiquetage du mot *que*.

En effet, nous avons remarqué que des erreurs d'étiquetage de certains mots fonctionnels ambigus induisent systématiquement de multiples erreurs de parsing. Le cas de *que* est particulièrement intéressant car très ambigu, mais la méthodologie présentée ici pourrait être appliquée à d'autres mots fonctionnels ainsi qu'à d'autres classes de mots facilement identifiables (ex. les nombres cardinaux). Nous examinons ici l'injection des connaissances linguistiques par deux moyens complémentaires : l'ajout de descripteurs riches lors de l'entraînement, et l'ajout de règles symboliques lors de l'analyse, qui imposent ou interdisent des décisions locales, contournant ainsi le modèle statistique. Notre approche ici, de correction de phénomènes spécifiques par l'injection d'informations symboliques, est similaire à certaines études précédentes, telles que Danlos (2005) pour le *il* impersonnel, et Jacques (2005) pour *que*. A la différence de ces études, qui considèrent des systèmes uniquement à base de règles, nous mettons l'accent ici sur les descripteurs riches, qui s'insèrent naturellement dans un système statistique robuste. Les règles symboliques sont utilisées uniquement en complément des descripteurs, pour des cas très précis et non ambigus.

2 Etiquetage morphosyntaxique

Dans Talismane, l'algorithme d'étiquetage morphosyntaxique fonctionne de gauche à droite. Ainsi, les descripteurs peuvent prendre en compte tous les tokens qui se trouvent à gauche et à droite du token à étiqueter, ainsi que les étiquettes déjà attribuées à sa gauche. Comme descripteurs de base, nous utilisons des descripteurs similaires à ceux décrits par Denis & Sagot (2012), faisant un usage massif d'un lexique, en l'occurrence le LeFFF (Sagot, 2010). En particulier, nous utilisons les descripteurs de base suivants : *W* la forme lexicale exacte, *P* l'étiquette attribuée au token (si son index < celui du token actuel) ou les étiquettes trouvées dans le lexique pour ce token (si son index $\geq$ celui du token actuel), *L* le lemme de ce token, pour une étiquette donnée, *U* si le token est inconnu dans le lexique, Sfx_n les n dernières lettres de la forme, $Pref_n$ les n premières lettres de la forme, *1st* si le token est le premier de la phrase, $Last$ si le token est le dernier de la phrase. Ces briques de base sont combinées en bigrammes et trigrammes pour les tokens à position -2, -1, 0, +1, +2 par rapport au token actuel. En vue de ce jeu de descripteurs, un descripteur plus "riche" est n'importe quel descripteur qui regarde plus loin que 2 tokens à gauche ou à droite, ou qui regroupe les tokens en classes d'équivalence à un niveau qui se trouve entre le lemme et l'étiquette morphosyntaxique. Dans la pratique, nous avons utilisé des descripteurs bien plus sophistiqués (décrits ci-après), qui mettent en oeuvre des combinaisons logiques complexes des informations de base.

Pour cette étude, notre corpus d'apprentissage est la partie française du corpus SPMRL (Seddah *et al.*, 2013), un corpus disponible en dépendances et construit à partir du French Treebank (FTB) (Abeillé *et al.*, 2003). Nous avons appliqué un pré-traitement aux mots composés, conservant uniquement les mots composés qui ne représentent pas une régularité syntaxique. Pour le corpus d'évaluation, en plus des parties *dev* et *test* du corpus SPMRL, nous utilisons les corpus Sequoia (Candito *et al.*, 2012) et un corpus des pages de discussion du Wikipedia français, FrWikiDisc, décrit dans Urieli (2013). Nous utilisons le jeu d'étiquettes décrit dans Crabbé & Candito (2008). Pour un modèle SVM linéaire construit avec les descripteurs ci-dessus, $\epsilon = 0,01$, $C = 0.5$ et un cutoff de 3 (nombre de fois qu'un descripteur doit apparaître pour être pris en compte), on a une exactitude de 96,58 sur SPMRL-dev, et 96,55 sur SPMRL-test. Toutes les données et les modèles sont disponibles sur simple demande, à l'exception du SPMRL français, qui nécessite un accord préalable.

3 Le cas de "que"

Les difficultés d'étiqueter le mot *que* ont déjà été explorées dans d'autres études, en particulier Jacques (2005), qui décrit les différents contextes dans lequel *que* est utilisé, et qui propose une méthode pour corriger l'étiquetage par un mélange de règles de surface et de corrections appliquées pendant l'analyse syntaxique.

Pour résumer, il y a six options principales pour le token *que* (et son équivalent abrégé *qu'*), annotées selon les normes d'annotation du corpus FTB avec 4 étiquettes différentes, comme illustré par les exemples suivants :

1. Conjonction de subordination (CS) : *Je pense qu'il a trop bu.*
2. Pronom relatif (PROREL) : *Il boit le vin que j'ai acheté.*
3. Pronom interrogatif (PROWH) : *Que buvez-vous ?*
4. Adverbe négatif (ADV) : *Je n'ai bu que trois verres.*
5. Adverbe exclamatif (ADV) : *Qu'il est bon, ce vin !*
6. Construction comparatif (CS) : *Il est plus bourré que moi.*

Le *que* d'une clivée est étiqueté PROREL pour un focus nominal argument du verbe. Quand le focus est un syntagme prépositionnel ou nominal circonstant, l'étiquetage du FTB est assez incohérent entre PROREL et CS.

	ADV	**CS**	**PROREL**	**PROWH**	**Total**	**Erreurs**
ADV	90	44	4	1	139	49
CS	37	1097	61	0	1195	98
PROREL	0	69	244	0	313	69
PROWH	0	4	2	23	29	6

TABLE 1 – Matrice de confusion de base pour *que*

Avec le modèle de base décrit dans le paragraphe précédent, la table 1 montre la matrice de confusion pour le mot *que* dans l'ensemble des corpus d'évaluation, où les lignes représentent la bonne étiquette et les colonnes représentent l'étiquette devinée. Nous avons donc au total 222 erreurs pour 1 676 occurrences, donc une exactitude de 86,75 %. Il est à noter que la confusion se trouve principalement entre CS and ADV d'une part, et entre CS et PROREL d'autre part. Nous traiterons, ci-dessous, chacun de ces cas séparément.

4 Des descripteurs ou des règles ?

Un **descripteur** (*feature* en anglais) spécifie l'information à extraire d'un contexte donné, qui pourra aider le classifieur à choisir la bonne étiquette du token dans ce contexte. Dans Talismane, un descripteur est défini par une expression qui combine des informations de base, soit par concatenation (pour les chaînes de caractères), soit par des opérations mathématiques (pour les nombres) ou logiques (pour les résultats booléens de type vrai/faux). Une **règle** est une expression booléene définie avec la même grammaire que les descripteurs. Si l'expression s'évalue à *vrai* dans un contexte donné, la règle peut soit imposer le choix d'une certaine étiquette pendant l'analyse, soit empêcher le système de choisir cette étiquette.

Par exemple, l'étiquette attribuée au mot précédant le *que* peut être utilisée comme descripteur. Si ce mot est un verbe à l'indicatif (V), e.g. *"il **faut** que…"*, alors on observe certaines tendances sur l'étiquetage du *que* : dans notre corpus d'apprentissage, sur 523 cas, 77 % sont des CS et 23 % des ADV. Un autre descripteur peut porter sur les étiquettes possibles du mot suivant le *que* dans un lexique externe de référence. Si ce mot est listé dans le lexique comme verbe à l'indicatif, e.g. *"l'exemple que **fournit** Dupont…"*, alors dans notre corpus d'apprentissage, sur 205 cas, 1 % des *que* sont des ADV, 20 % des CS, 72 % des PROREL et 7 % des PROWH. Ces informations vont être combinées avec des dizaines d'autres descripteurs pour aider le modèle probabiliste à construire une distribution de probabilités des étiquettes pour un cas donné de *que*.

On peut aussi être amené à définir une règle déterministe : e.g. si on a une structure de type *"ne V que"*, alors on oblige le système à attribuer l'étiquette ADV. Cette règle prend priorité sur le modèle probabiliste qui ne sera même pas consulté. Un autre type de règle est la règle négative : e.g. si *que* est le premier mot d'une phrase, alors on empêche le système d'attribuer l'étiquette PROREL. Dans ce cas, le modèle probabiliste va utiliser tous les descripteurs pour définir une distribution de probabilités des étiquettes, mais l'étiquette PROREL sera supprimée de cette distribution avant que le système ne choisisse l'étiquette la plus probable.

Un descripteur cherche, par nature, à capter des régularités dans le corpus d'apprentissage qui peuvent se généraliser à d'autres corpus. Il est donc limité aux régularités qui se trouvent dans ce corpus, même si elles peuvent être décrites à l'aide de ressources externes pour les rendre plus généralisables (ex. un lexique qui remplace la forme lexicale par son lemme). Par contre, une règle cherche à traduire directement les connaissances linguistiques du concepteur du système, surtout pour des phénomènes sous représentés dans le corpus d'apprentissage. Elle permet donc au système statistique

d'aller au delà des informations qui lui sont directement accessibles. Puisque celle-ci est appliquée uniquement au moment de l'analyse, elle peut aussi traduire des connaissances spécifiques au corpus qu'on est en train d'analyser.

Les descripteurs serviront à alimenter le classifieur (ex. SVM), qui va appliquer sa "magie noire" statistique pour donner plus ou moins de poids à chaque descripteur pour chaque étiquette, selon les occurrences trouvées dans le corpus d'apprentissage. Les descripteurs peuvent donc se contredire et se chevaucher. Ils décrivent des tendances : si X est vrai, alors l'étiquette sera plus probablement Y que Y'. Les règles cherchent, par contre, à décrire des vérités absolues : si X est vrai, alors l'étiquette doit être (ou ne peut pas être) Y. Les descripteurs sont par conséquent beaucoup plus puissants que les règles, car à la différence de celles-ci, ils ne sont pas contraints à viser un phénomène très spécifique et non ambigu. Néanmoins, vu le coût de construction d'un corpus annoté, les corpus sont forcément très lacunaires en informations. Ce sont ces informations que les règles vont cibler.

5 "Que" comme adverbe négatif

En termes de descripteurs ciblés, nous traiterons d'abord le cas de *que* en tant qu'adverbe négatif. Notre méthodologie itérative consiste à :

1. Analyser les erreurs dans le corpus *dev* et concevoir des descripteurs utiles.
2. Écrire ces descripteurs dans la syntaxe de Talismane.
3. Projeter ces descripteurs sur le corpus *train*, et examiner les co-occurrences avec chaque étiquette, surtout celles avec une étiquette inattendue. Nous cherchons à inclure le maximum de résultats tout en maximisant le déséquilibre entre les étiquettes. Revenir à l'étape 2 si nécessaire.
4. Entraîner le modèle avec les descripteurs ciblés, et évaluer. Revenir à l'étape 1 si nécessaire.

5.1 Analyse d'erreurs : adverbe négatif

Dans le corpus SPMRL *dev*, la plupart des erreurs ressemblent au cas suivant , ou *que* est étiquetté à tort comme `CS` :

Exemple 5.1 *Mais cela **ne** représente dans cette mouture, pour un couple avec deux enfants, **qu'**/ADV une prime maximale.*

Dans ce cas, reconnaître *que* comme adverbe négatif revient à chercher une occurrence de *ne* plus tôt dans la même phrase. Il n'y a pas de limitation inhérente de distance car, comme on voit dans l'exemple 5.1, plusieurs syntagmes prépositionnels peuvent séparer les deux particules. Par contre, une autre particule négative peut compléter le *ne*, ce qui rend le *que* ambigu, comme dans les deux exemples suivants :

Exemple 5.2 *Pour cela, il **n'**est **pas** question **que**/CS le zloty, la monnaie polonaise, soit "l'ancre de la stabilité" de l'économie polonaise.*

Exemple 5.3 *…qui, faute de volonté politique, **ne** fut **jamais que**/ADV la caricature du système français.*

Cette ambiguïté existe uniquement pour les verbes qui sous catégorisent un objet direct en *que*, tels que *dire* ou *penser*. Cependant, le corpus *train* contient 245 verbes différents qui répondent à ce critère. Les cas ambigus où le *que* suit une autre particule négative étant assez rares, nous avons décidé de ne pas utiliser la sous catégorisation dans nos descripteurs.

5.2 Liste de descripteurs : adverbe négatif

La prochaine étape consiste à écrire ces descripteurs dans la syntaxe de Talismane, et les projeter sur le corpus *train*. Après affinage pour étendre la portée des descripteurs tout en éliminant des cas non voulus, nous avons défini la liste suivante. Les nombres représentent le nombre d'occurrences dans le corpus *train*.

Descripteur 5.1 ***Ne* précédent sans autre particule négative** : le *que* est précédé par un *ne* sans autre particule négative entre les deux. En plus le *ne* n'est pas lui-même précédé par {*personne*, *rien*, *aucun/e*, *nul/le*}, afin d'exclure des phrases comme *"Personne ne sait que je mange ici."*.

Nous avons 345 cas en tout, dont 312 `ADV` : *"Ils **n'**en comprendront le sens **que**/ADV bien plus tard"* ; et 32 `CS`. Parmi les `CS`, il y a beaucoup d'erreurs d'annotation—les autres sont des phrases où la particule *ne* n'est pas completé par une autre particule négative, dans des expressions de type *moins ADJ qu'on ne…* : *"L'Amérique, moins superficielle qu'on **ne** l'imagine parfois, a entrepris une réflexion sur son identité bien avant **que**/CS […]"* ; ou en modifiant le verbe *pouvoir* : *"[…] **ne** peuvent ainsi éviter **que**/CS, en la matière, l'histoire ne se repète"*.

Descripteur 5.2 Pas de *ne* précédent : il n'y a pas de *ne* précédent le *que*.

Nous avons 2608 cas en tout, dont 1941 `CS`, 622 `PROREL`, 26 `PROWH` et 19 `ADV`, dont 10 sont des erreurs d'annotation, où un *que* comparatif est annoté comme adverbe, 5 sont des adverbes exclamatifs : *"Mais pour parvenir à cela, **que**/ADV d'esprits à convaincre en France et plus encore au-dehors !"* ; et 1 est une phrase "informelle" ou l'auteur a laissé tomber le *ne* : *"Il lui manque **que**/ADV le sac à main de Maggie"*.

Nous avons ajoutés deux descripteurs supplémentaires pour aider l'étiqueteur dans des cas où il y a une autre particule négative entre le *ne* et le *que* :

Descripteur 5.3 ***Que* négatif possible** : y a-t-il un *ne*, *ne pas* ou *ne plus* plus tôt dans la phrase, sans prendre en compte d'autres particules négatives. Ce descripteur couvre tous les cas où un *que* négatif est possible. C'est une version moins exclusive du descripteur 5.1. On trouve 363 `ADV`, 218 `CS` et 47 `PROREL`. Pour ces deux derniers, la grande majorité sont des cas où le *ne* est complété par une autre particule négative.

Descripteur 5.4 Combinaison de particules négatives à courte distance : nous avons remarqué que, dans le corpus d'apprentissage, le *que* se combine avec une autre particule négative uniquement si leur distance est petite (≤ 6 tokens). Ce descripteur s'évalue donc à `vrai` si la distance est ≤ 6, à `faux` si la distance est plus grande, et à rien du tout s'il n'y a pas de particule négative entre le *ne* et le *que*.

Pour la distance courte, sur 184 cas, nous avons 119 `CS`, 17 `PROREL` et 47 `ADV`, ce qui représente plus d'un quart des cas : *"Les spéculateurs sont désormais certains que la dévaluation **n'**est **plus qu'**/ADV une question de jours"*. Pour la distance longue, sur 146 cas, nous avons 101 `CS` : *"Si cela **n'**était **pas** possible, les Onze poursuivraient leur chemin sans perdre l'espoir **que**/CS cela se ferait plus tard"* ; 43 `PROREL` et uniquement 2 `ADV` : *"Il **ne** restait **plus** au président du groupe socialiste de l'Assemblée nationale, dans ces conditions, **qu'**/ADV à négocier la fusion de son texte avec celui de MM Jospin et Delebarre"*.

6 "Que" comme pronom relatif

A la différence du *que* en tant qu'adverbe négatif, où la présence d'un *ne* précédent est un indicateur de surface fort, il n'y a pas d'indicateur de surface simple pour distinguer le *que* pronom relatif du *que* conjonction de subordination, étant donné le peu d'informations disponibles à l'étape de l'étiquetage morphosyntaxique.

6.1 Analyse d'erreurs : pronom relatif

Suivant la méthodologie décrite dans le paragraphe 5, nous analysons les erreurs du corpus *dev* pour identifier des descripteurs utiles.

Exemple 6.1 (…) *la Commission des opérations de bourse (COB) a annoncé le 14 janvier **qu'**/CS elle saisit la justice […]*

Cet exemple est annoté PROREL plutôt que CS. Certains descripteurs sautent aux yeux : d'abord *annoncer* est parmi les verbes qui sous catégorisent un objet direct avec *que*. De plus, le verbe transitif *saisir* a déjà un objet direct (*justice*), ce qui exclut généralement un pronom relatif. Finalement, noter que l'ambiguïté entre PROREL et CS existe uniquement quand il y a un nom qui peut servir d'antécédent entre le verbe précédent et le *que*, dans ce cas *janvier*. La nature de ce nom est un indicateur : les expressions de temps, dont les noms des mois, sont très souvent des circonstants. Ils remplissent rarement l'argument d'objet direct, et sont rarement modifiés par une proposition relative.

Exemple 6.2 *Le gouvernement va présenter dans un délai de trois mois les dispositions **qu'**/PROREL il entend retenir [...]*

Cet exemple est annoté CS plutôt que PROREL. C'est le cas contraire de l'exemple précédent : le verbe *présenter* a déjà un objet direct (*dispositions*) et ne sous catégorise pas un objet direct avec *que*, alors que le verbe transitif *retenir* n'a pas d'objet direct qui le suit.

Nous voyons ici l'importance de reconnaître les verbes qui sous catégorisent avec *que*. Comme mentionné précédemment, le corpus *train* en contient 245. Nous avons choisi manuellement 152 de ces verbes qui nous semblaient les plus aptes à préférer cette sous catégorisation.

Exemple 6.3 *Le fait **qu'**/CS ils aient accepté de reprendre les pourparlers est interprété de façon positive.*

Ici nous avons d'autres indicateurs : certains noms introduisent des propositions subordonnées (ex. *fait*), et le subjonctif (*aient*) indique généralement qu'on a affaire à une subordonnée indépendante plutôt que relative.

6.2 Liste de descripteurs : pronom relatif

Après la projection des descripteurs sur le corpus *train* et affinage, nous avons retenu les descripteurs suivants :

Descripteur 6.1 Structure coordonnée : Si le *que* actuel suit directement une conjonction de coordination, chercher l'étiquette du *que* précédent. De même, si le *que* actuel suit une virgule, chercher l'étiquette du *que* précédent, du moment où il existe un *que* plus tard dans la phrase qui suit une conjonction de coordination. Si l'étiquette précédente est CS (total 104 cas), nous avons 102 CS : *"Or chacun est conscient **qu'**/CS il n'y a aucune vérification de ces acquis et **que**/CS la rétribution est automatique"* ; 1 PROREL : *"Encore faudrait-il **que**/CS, pour faire passer la pilule des réformes nécessaires—et **que**/PROREL beaucoup d'Italiens risquent de trouver plus amère que prévu [...]"* ; et 1 *PROWH*. Si l'étiquette précédente est PROREL (total 8 cas), nous avons 2 CS et 6 PROREL.

Descripteur 6.2 Après nom explicatif : Le *que* suit-il un des mots {*assurance, certitude, doute, enseigne, espoir, fait, fois, idée, point, prétexte, preuve, principe*} ? Les cas précédés par la locution *c'est* ont été exclus. Résultat : 38 CS.

Descripteur 6.3 Verbe précédent sous catégorise avec *que* : Chercher le verbe précédent (en faisant attention de sauter les participes passés modifieurs de noms). On s'intéresse uniquement aux cas où il y a un nom entre le verbe et le *que*, qui peut servir d'antécédent. Est-ce que ce verbe sous catégorise avec *que* ? Pour les cas où le verbe précédent sous catégorise avec *que* (total 98 cas), nous avons 50 CS : *"Helmut Kohl a **annoncé** à l'automne **que**/CS des hausses d'impôts seraient nécessaires en 1994"* ; et 48 PROREL : *"Il a toutefois **refusé** à Mr Vernay les 100 000 francs de dommages et intérêts **que**/PROREL celui-ci réclamait"*. Les deux étiquettes sont donc distribuées de façon à peu près égale. Pour le cas contraire (total 126 cas), nous avons 113 PROREL, et uniquement 12 CS, dont 10 erreurs d'annotation.

Descripteur 6.4 Le verbe qui précède a un objet direct : Le verbe précédant le *que* est-il suivi directement d'un déterminant et d'un nom (ou d'un déterminant, d'un adjectif et d'un nom), en dehors des noms représentant les expression de temps (ex. *la semaine dernière*) ? On enlève les cas où le *que* suit directement un nom "explicatif" du descripteur 6.2. Sur 63 cas, nous avons 56 PROREL : *"En revanche, la CGT **dénonce un texte qu'**/PROREL elle juge "décrédibilisé par le manque de moyens""* ; et 7 CS : *"Nous avons **obtenu l'assurance** du premier ministre **que**/CS la suppression du recours [...]"*, dont 6 sont des erreurs d'annotation.

Descripteur 6.5 Le verbe qui précède sous catégorise avec à + personne + que : Le verbe précédent le *que* est-il dans l'ensemble {*annoncer, certifier, ...*} qui sous catégorise des structures comme *"annoncer à ses parents qu'on se marie"*, et est-il suivi de la préposition *à* ? Résultats : 23 CS.

Descripteur 6.6 Le verbe qui précède sous catégorise avec un objet direct + que : Le verbe précédent le *que* est-il dans l'ensemble {*assurer, avertir, ...*} qui sous catégorise des structures comme *"assurer ses parents qu'on se marie"* ? Résultats : 26 CS.

Descripteur 6.7 Le verbe qui suit a un objet direct : Le verbe suivant le *que* est-il suivi directement d'un déterminant et d'un nom (ou d'un déterminant, d'un adjectif et d'un nom), en dehors des noms représentant les expressions de temps ? Résultats : 93 CS.

Descripteur 6.8 *Que* suivi directement d'un verbe : Le *que* est-il suivi directement d'un verbe ? On s'attend ici surtout à des PROREL et des PROWH. Pour 116 cas, nous avons 105 PROREL : *"L'exemple **que/PROREL fournit** Sombart est particulièrement éclairant"* ; 5 PROWH : *"Si vous pouviez changer le monde, **que/PROWH feriez**-vous ?"* ; et 6 CS, tous des verbes intransitifs avec inversion du sujet, dont 4 sont des formes subjonctives du verbe *être* : *"Ne souhaitant **que/CS soit** envisagée l'hypothèse d'une dévaluation du franc [...]"*.

Le même descripteur sans le verbe *être* donne 89 PROREL, 5 PROWH et 2 CS.

Descripteur 6.9 *Que* suivi d'un verbe réfléchi : Le *que* est-il suivi d'un verbe réfléchi à la troisième personne (à l'exception de certains verbes qui prennent un objet direct en plus du clitique réfléchi, tels *"se poser une question"*) ? Résultats : 12 CS.

Descripteur 6.10 *Que* suivi d'un verbe subjonctif : Le *que* est-il suivi d'un verbe d'une forme clairement subjonctive ? Puisqu'on regarde à droite du token actuel, on n'a pas encore les étiquettes morphosyntaxiques, et on compte sur le lexique pour nous indiquer les tokens qui peuvent être des verbes. Du coup, on s'est retrouvé au départ avec le nom *émissions* comme imparfait du subjonctif du verbe *émettre*. Nous avons donc éliminé les cas où le token avait aussi une étiquette non verbale dans le lexique. Résultat : 80 CS : *"Faut-il encore **que/CS** l'ambiance non seulement le **permette** mais aussi le favorise"* ; et 1 PROREL : *"Faut-il en conclure que le mieux **qu'/PROREL** on **puisse** attendre, c'est le chacun-pour-soi ?"*

Descripteur 6.11 Après clivée : Le *que* suit-il l'expression *c'est*, indiquant une clivée potentielle (typiquement annoté par un PROREL) ? Ayant remarqué que plus la locution *c'est* est proche, plus la clivée est probable, nous avons défini 4 descripteurs, qui cherchent le *c'est* à une distance de 5, 10 et 20 tokens avant le *que*, et sans limite. Les résultats sont pour une distance sans limite : 90 CS et 95 PROREL. Pour une distance de 20 : 79 CS et 90 PROREL. Pour une distance de 10 : 56 CS et 82 PROREL. Pour une distance de 5 : 34 CS et 62 PROREL.

Descripteur 6.12 Etre ADJ que : Trouver les structures de type *"il est probable que..."*. Résultats : 47 CS.

Descripteur 6.13 Existence d'un nom antécédent ? Y'a-t-il un nom (en dehors des expressions de temps) entre le verbe qui précède et le *que* ? Si le *que* est clairement un deuxième coordonné, on cherche avant le premier coordonné. Ici on considère les réponses à la fois positives et négatives. Résultats positifs (cas classique d'ambiguïté CS/PROREL) : 485 CS et 297 PROREL. Résultats négatifs : 1217 CS et 7 PROREL (dont la plupart sont des erreurs dans la reconnaissance du premier coordonné).

Descripteur 6.14 Expression de temps ? Le nom le plus proche avant le *que* fait-il partie d'une expression de temps (ex. *la semaine dernière*) ? Résultats : 53 CS : *"M. Tchechinski a estimé **dimanche que/CS** la Russie ne manquerait pas de pain cet hiver"* ; et 1 PROREL : *"Voici la **nuit que/PROREL** nous avons attendue toute l'année"*.

Descripteur 6.15 Phrase débutant par *que* : la phrase commence-t-elle par *que* (eventuellement précédé d'une conjonction de coordination) ? Sur 51 cas, nous avons 4 ADV, tous exclamatifs : *"**Qu'/ADV** il était insouciant, le mois de janvier 1992"* ; 28 CS, presque tous des phrases incomplètes : *"**Qu'/CS** ils sont "prescripteurs", comme disent les professionnels"* ; 1 PROREL dans une phrase incomplète : *"**Qu'/PROREL** elle n'aime guère voir rapprocher de celui des sociétés concurrentes"* ; et 18 PROWH : *"**Que/PROWH** fait-il de l'excédent de ses revenus ?"*

Descripteur 6.16 Pas de verbe avant le *que* : Si le *que* suit un nom, et qu'il n'y a pas de verbe plus tôt dans la phrase. On exclut les mots explicatifs ci-dessus. Résultats : 100 PROREL : *"Une puissance **qu'/PROREL** elle n'entend partager avec nul autre"* ; et 8 CS : *"Mais quelle prestance **que/CS** celle de l'homme-terminal !"*

Descripteur 6.17 Plus que : Le mot *que* suit-il un quantifieur {*plus, moins, davantage, autant, différent, même, tel, ...*} ? Résultats : 118 CS et 4 PROREL (tous des erreurs d'étiquetage avec *tel que*).

Descripteur 6.18 Plus ADJ que : Le mot *que* suit-il un adverbe comparatif {*plus, moins, davantage, autant, ...*} et un adjectif ? Résultats : 139 CS : *"Cela est **plus motivant que/CS** d'avoir des salariés sous-employés en temps et en compétences"* ; et 1 PROREL : *"L'un des problèmes les **plus graves qu'/PROREL** affronte l'université est celui du premier cycle, avec son considérable taux d'échec."*

Descripteur 6.19 Expression comparative complexe : expressions de type *plus de X que de Y* ou *aussi X que Y*. Résultats : 81 CS : *"C'est **moins** une mode **qu'/CS** un uniforme"* ; et 3 PROREL : *"C'est un organisme du **même** genre **que/PROREL** l'on veut créer au bénéfice de l'Europe tout entière."*

7 Résultats pour les descripteurs ciblés

	ADV		CS		PROREL		PROWH		Total	Erreurs	
ADV	133	(+43)	6	(-38)	0	(-4)	0	(-1)	139	6	(-43)
CS	10	(-27)	1135	(+38)	50	(-11)	0	(-1)	1195	60	(-38)
PROREL	0		52	(-17)	261	(+17)	0		313	52	(-17)
PROWH	0		0	(-4)	4	(+2)	25	(+2)	29	4	(-2)

TABLE 2 – Matrice de confusion pour *que* avec les descripteurs ciblés

La table 2 montre la matrice de confusion pour *que* après l'ajout des descripteurs riches. Les résultats sont considérablement améliorés pour toutes les catégories, mais plus particulièrement pour la confusion entre ADV et CS. En tout, nous avons supprimé 45 % des erreurs, passant d'une exactitude de 86,75 % à 92,72 %. Les résultats sont hautement significatifs, avec 139 nouvelles corrections pour 29 nouvelles erreurs (test de McNemar, p-valeur < 0,001). Pourtant, ces gains ont un prix : la vitesse d'analyse. Dans la version de base, on étiquette 1 million de mots en 6m48s. Avec les descripteurs ciblés, cela prend 1,5 fois plus de temps : 10m09s.

8 Les règles

Dans le paragraphe précédent, certains cas n'ont pas été corrigés même quand les descripteurs riches ajoutaient du poids à la bonne étiquette : les descripteurs riches semblaient noyés dans un océan de descripteurs plus pauvres et génériques, ce qui les empêchait de faire pencher la balance en faveur de la bonne étiquette.

En analysant les erreurs restantes, nous avons identifié certaines règles qui nous semblaient généralisables. Vu la rareté des phénomènes qui peuvent être ciblés par des règles, nous avons examiné ici les erreurs dans tous les corpus sauf SPMRL *test* et EMEA *test*, à la différence de l'expérience avec les descripteurs, où seulement les erreurs de SPMRL *dev* avaient été examinées. Nous avons retenu les règles suivantes :

Règle 8.1 Étiqueter PROWH si la phrase se termine par un point d'interrogation, si *que* est le premier mot de la phrase (hors conjonctions de coordination), et si *que* est directement suivi par un verbe à l'indicatif ou l'infinitif (avec ou sans les clitiques *en* ou *se*). Exemple : *"Et que/PROWH se passera-t-il si un seul syndicat signe un accord de ce type contre l'avis des autres ?"*

Règle 8.2 Étiqueter PROWH le premier *que* dans *qu'est-ce que*.

Règle 8.3 Étiqueter CS dans les locutions de type *attendre/veiller/tenir à ce que*, *n'empêche que*, *dommage que*, *avoir honte à ce que*, *le/du/au fait que*, *une fois que*.

Règle 8.4 Étiqueter CS pour toute locution de type *être ADJ que*, tel que "*il est probable que*", sauf si l'expression est précédé de *ne*.

Règle 8.5 Étiqueter PROREL dans *ceux/celui/celle/celles/quoi/qui/quel/quelle/quels/quelles/où que* et dans l'expression *tout ce que*.

Règle 8.6 Étiqueter PROREL dans les locutions *c'est NPP que* ou *c'est DET NC que*

Règle 8.7 Étiqueter ADV pour toute locution verbale de forme *ne V que*, où il n'existe pas le mot *rien, personne, aucun, aucune, ni* ou *jamais* plus tôt dans la phrase.

Règle 8.8 Ne pas étiqueter PROREL si *que* est le premier mot de la phrase.

Règle 8.9 Ne pas étiqueter PROREL si *que* suit un verbe directement, ou s'il est séparé du verbe uniquement par un commentaire entouré par des virgules, des tirets ou des parenthèses. La dernière condition est un exemple d'une règle visant un corpus spécifique, en l'occurrence le corpus Europarl de Séquoia, qui contient beaucoup de phrases du type *"Je sais, Madame la Présidente, **que**/CS vous êtes déjà intervenue [...]"* où le *que* a été étiqueté PROREL.

Règle 8.10 Ne pas étiqueter PROREL si *que* suit les mots *reprises, tous, toutes, toute, tout, soi*

Règle 8.11 Ne pas étiqueter ADV sauf s'il y a un *ne* plus tôt dans la phrase, ou si *que* est le premier mot de la phrase.

	ADV	CS	PROREL	PROWH	Total	Erreurs
ADV	134 (+1)	5 (-1)	0	0	139	5 (-1)
CS	10	1149 (+14)	36 (-14)	0	1195	46 (-14)
PROREL	0	48 (-4)	265 (+4)	0	313	48 (-4)
PROWH	0	0	2 (-2)	27 (+2)	29	2 (-2)

TABLE 3 – Matrice de confusion pour *que* avec les règles

La table 3 montre les résultats après l'ajout des règles pour tous les corpus *dev* et *test*, avec, entre parenthèses, les gains par rapport au modèle des descripteurs riches. Sans surprise, les résultats sont positifs, puisqu'on a visé des erreurs trouvées dans ces mêmes corpus *dev*. Il nous reste 101 sur 222 erreurs, soit une réduction de 55 % du taux d'erreur, avec une exactitude de 93,97 %. Au niveau de la significativité, nous avons 21 nouvelles corrections pour 0 nouvelles erreurs. Pourtant, il y a un risque de suradéquation des règles aux corpus évalués, les corpus *test* étant trop petits pour mesurer l'impact plus global des règles. Du coup, nous avons testé les mêmes règles sur des corpus non annotés, comparant les différences entre les analyses avec et sans règles. Nous avons donc analysés, avec et sans règles, 200 000 mots de chacun des corpus suivants :

– Est Républicain : le journal régional Est Républicain de l'année 2003, disponible sur le site du CNRTL [2]
– Leximedia : une collection d'articles concernant la campagne présidentielle 2007, extraits des journaux nationaux Le Monde, Libération et Le Figaro, et préparé par le laboratoire CLLE-ERSS [3]
– Frantext : des textes littéraires français du 20ème siècle [4]
– Revues.org : une collection d'articles scientifiques dans les sciences sociales [5]

Au total, il y a uniquement une différence tous les 8 500 mots, mais avec un bilan très positif : 46 corrections pour 5 erreurs dans les 51 premières différences. Les erreurs pourraient être éliminées facilement si on affinait les règles. Les corrections les plus intéressantes concernent les commentaires qui séparent le *que* du verbe qui le gouverne. La règle corrige l'étiquette PROREL en CS, comme dans la phrase suivante :

Exemple 8.1 *Je **conteste**, en tant que père de famille, **que**/CS l'on vienne me dire que l'argent est le corollaire du succès.*

2. http://www.cnrtl.fr/corpus/estrepublicain/
3. http://redac.univ-tlse2.fr/applications/leximedia2007.html
4. http://www.frantext.fr
5. http://www.revues.org/

9 Conclusions et perspectives

Nous avons voulu, dans la présente étude, démontrer l'intérêt d'injecter des connaissances linguistiques riches dans un système statistique. En conclusion, il est clair que les descripteurs riches, spécifiques à une langue donnée, permettent de corriger un grand nombre d'erreurs dans le cas de l'étiquetage morphosyntaxique des mots fonctionnels, avec une réduction de 45 % du taux d'erreur pour le mot *que*. L'application supplémentaire de règles très spécifiques permet d'atteindre une réduction totale de 55 %.

Reste la question de la facilité de maintenance des systèmes basés sur ce type d'information plus riche. Par rapport aux systèmes "rationalistes" qui fonctionnent uniquement à base de règles formelles et cherchent à décrire une langue de façon complète, les systèmes "empiriques" statistiques, à base d'apprentissage supervisé, ont le grand avantage de laisser le travail de description linguistique au corpus d'apprentissage. Du coup, la complexité de la langue peut être décrite dans un guide d'annotateur, au lieu d'être codée de façon formelle au sein du logiciel. En plus, on se contente de décrire les cas présents dans un corpus donné, mettant ainsi l'accent sur les cas les plus courants. Cette abstraction de la complexité linguistique rend la maintenance du système beaucoup plus simple. Cela reste-t-il vrai dans la présente étude ? Dans le cas des descripteurs ciblés la réponse est "oui" : autant l'écriture et l'affinage des descripteurs peuvent s'avérer long et complexe, autant la maintenance du système à long terme est simple, puisque le modèle probabiliste ajuste automatiquement le poids de chaque descripteur au fur et à mesure que d'autres descripteurs sont ajoutés, où que d'autres données d'apprentissage deviennent disponibles. Pour les règles, la réponse est plus complexe. Il est important de baser les règles uniquement sur des erreurs effectivement rencontrées dans le corpus de développement, et que l'on peut décrire de façon non ambigüe. Ceci réduit considérablement le nombre de règles, qui sont là uniquement pour compléter le système dans certains cas bien définis : la plupart du travail continue à être fait par les descripteurs.

Les perspectives sont nombreuses : tout d'abord, il y a la question de la vitesse d'analyse. Bien que l'étiquetage morpho-syntaxique de Talismane soit assez rapide après l'ajout des descripteurs et des règles (5 millions de mots / heure), il est plus lent que dans la version de base (9 millions de mots / heure). Nous avons choisi pour l'instant de mettre tous les descripteurs dans un fichier de configuration. L'avantage est de séparer complètement les descripteurs du code source, permettant ainsi à un linguiste d'inventer de nouveaux descripteurs sans l'aide d'un informaticien. L'inconvénient est que certaines opérations (ex. recherche en avant et en arrière) sont répétées de nombreuses fois, alors qu'elles pourraient être codées de façon bien plus efficace dans un langage informatique compilé et plus expressif. Il serait donc intéressant de mesurer le gain de vitesse en codant ces mêmes descripteurs dans un langage compilé.

Il serait aussi souhaitable d'effectuer une analyse plus fine des erreurs qui restent après l'application des descripteurs et des règles : y a-t-il encore une possibilité de diminuer le taux d'erreur ?

Ensuite, nous souhaitons appliquer la même méthodologie à d'autres mots fonctionnels, tels que "de/des/du" et "soit". Finalement, nous souhaitons tester une méthodologie semblable sur les erreurs du parseur, pour voir si les descripteurs riches et les règles peuvent être aussi efficaces dans le contexte plus compliqué du parsing par transitions.

Remerciements

Je tiens à remercier les relecteurs anonymes pour leurs commentaires et suggestions. Je tiens aussi à remercier l'équipe de l'axe CARTEL à CLLE-ERSS pour leur soutien pendant ce travail.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building a treebank for French. In A. ABEILLÉ, Ed., *Treebanks*. Kluwer.

B. BIGI, Ed. (2014). *Actes de TALN 2014 (Traitement automatique des langues naturelles)*, Marseille. ATALA, LPL.

CANDITO M., SEDDAH D. *et al.* (2012). Le corpus sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *TALN 2012-19e conférence sur le Traitement Automatique des Langues Naturelles*.

CRABBÉ B. & CANDITO M. (2008). Expériences d'analyses syntaxique statistique du français. In *TALN 2008- conférence sur le Traitement Automatique des Langues Naturelles* : ATALA.

DANLOS L. (2005). Ilimp : Outil pour repérer les occurrences du pronom impersonnel il. In *Actes de la 12ème conférence sur le Traitement Automatique des Langues Naturelles (TALN'2005)*, p. 123–132, Dourdan, France.

DENIS P. & SAGOT B. (2012). Coupling an annotated corpus and a lexicon for state-of-the-art pos tagging. *Language Resources and Evaluation*, **46**(4), 721–736.

JACQUES M.-P. (2005). Que : la valse des étiquettes. In *Actes de la 12ème conférence sur le Traitement Automatique des Langues Naturelles (TALN'2005)*, p. 133–142, Dourdan, France.

KÜBLER S., MCDONALD R. & NIVRE J. (2009). *Dependency parsing*. Morgan & Claypool Publishers.

SAGOT B. (2010). The lefff, a freely available and large-coverage morphological and syntactic lexicon for french. In *7th international conference on Language Resources and Evaluation (LREC 2010)*.

SEDDAH D., TSARFATY R., KÜBLER S., CANDITO M., CHOI J., FARKAS R., FOSTER J., GOENAGA I., GOJENOLA K., GOLDBERG Y., GREEN S., HABASH N., KUHLMANN M., MAIER W., NIVRE J., PRZEPIORKOWSKI A., ROTH R., SEEKER W., VERSLEY Y., VINCZE V., WOLIŃSKI M., WRÓBLEWSKA A. & VILLEMONTE DE LA CLÉRGERIE E. (2013). Overview of the spmrl 2013 shared task : A cross-framework evaluation of parsing morphologically rich languages. In *Proceedings of the 4th Workshop on Statistical Parsing of Morphologically Rich Languages : Shared Task*, Seattle, WA.

URIELI A. (2013). *Robust French syntax analysis : reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. PhD thesis, Université de Toulouse II le Mirail.

URIELI A. & TANGUY L. (2013). L'apport du faisceau dans l'analyse syntaxique en dépendances par transitions : études de cas avec l'analyseur Talismane. In *Actes de la 20e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2013)*, p. 188–201, Les Sables d'Olonne, France.

ZHANG Y. & NIVRE J. (2011). Transition-based dependency parsing with rich non-local features. In *ACL (Short Papers)*, p. 188–193.

Jouer avec des analyseurs syntaxiques

Éric Villemonte de la Clergerie INRIA - Rocquencourt - B.P. 105
78153 Le Chesnay Cedex, FRANCE
Eric.De_La_Clergerie@inria.fr

Résumé. Nous présentons DYALOG-SR, un analyseur syntaxique statistique par dépendances développé dans le cadre de la tâche SPRML 2013 portant sur un jeu de 9 langues très différentes. L'analyseur DYALOG-SR implémente un algorithme d'analyse par transition (à la MALT), étendu par utilisation de faisceaux et de techniques de programmation dynamique. Une des particularité de DYALOG-SR provient de sa capacité à prendre en entrée des treillis de mots, particularité utilisée lors de SPMRL13 pour traiter des treillis en Hébreu et reprise plus récemment sur des treillis produits par SXPIPE pour le français. Disposant par ailleurs avec FRMG d'un analyseur alternatif pour le français, nous avons expérimenté un couplage avec DYALOG-SR, nous permettant ainsi d'obtenir les meilleurs résultats obtenus à ce jour sur le French TreeBank.

Abstract. We present DYALOG-SR, a statistical dependency parser developed for the SPRML 2013 shared task over 9 very different languages. DYALOG-SR implements a shift-reduce parsing algorithm (a la MALT), extended with beams and dynamic programming techniques. One of the specificities of DYALOG-SR is its ability to handle word lattices as input, which was used for handling Hebrew lattices and more recently French ones produced by SXPIPE. Having access to FRMG, an alternative parser for French, we also tried a coupling with DYALOG-SR, providing us the best results so far on the French TreeBank

Mots-clés : Analyse syntaxique, Analyse syntaxique par dépendances, faisceaux, Programmation Dynamique, Treillis de mots, Couplage d'analyseurs.

Keywords: Parsing, Dependency Parsing, Beams, Dynamic Programming, Word Lattice, Parser coupling.

1 Introduction

Nous présentons diverses expériences d'analyse syntaxique pour le français menées avec l'analyseur statistique DYALOG-SR (Villemonte De La Clergerie, 2013a). Initialement développé pour participer à la campagne organisée en marge de SPMRL 2013 (Seddah *et al.*, 2013), cet analyseur en dépendances a été testé sur 9 langues très diverses, comme l'hébreu, le hongrois ou le coréen, et a terminé second dans sa catégorie. S'appuyant sur une stratégie d'analyse par transitions (à la MALT (Nivre, 2003)), DYALOG-SR utilise de plus la programmation dynamique pour gérer des choix non-déterministes au travers de faisceaux (*beams*). Mais la principale originalité de DYALOG-SR réside dans sa capacité à traiter en entrée des treillis de mots, pouvant représenter des ambiguïtés lexicales mais également des ambiguïtés de segmentation. Cette capacité a été utilisée dans le cadre de SPMRL sur des treillis en hébreu.

À ce stade, une première expérience préliminaire a consisté à entraîner et à évaluer DYALOG-SR sur la version en dépendances du French TreeBank (FTB) (Candito *et al.*, 2010a), de manière à pouvoir le comparer avec les autres systèmes évalués sur ce même treebank (Candito *et al.*, 2010b; Urieli & Tanguy, 2013; Le Roux *et al.*, 2012). Il nous a paru intéressant de tester également DYALOG-SR sur les treillis de mots produits par le segmenteur SXPIPE (Sagot & Boullier, 2008), en particulier pour vérifier son comportement sur les treillis pour une autre langue que l'hébreu. De fait, nous avons du affiner certains détails de l'algorithme, en particulier pour une meilleure prise en compte des mots en lecture avant (*lookahead*).

Par ailleurs, au travers de FRMG (de La Clergerie, 2005b), nous disposons d'un autre analyseur pour le français, basé sur une grammaire linguistique, de bonne qualité et de large couverture, capable de produire des analyses respectant le schéma d'annotation du FTB (Villemonte De La Clergerie, 2013b). Ceci nous a incité à tenter une expérience de couplage entre DYALOG-SR et FRMG, en utilisant les résultats de FRMG comme traits de guidage pour DYALOG-SR. Cette manipulation, finalement très simple à mettre en oeuvre, s'est révélée fructueuse et assure les meilleurs performances, à notre connaissance, obtenues à ce jour sur le FTB.

Dans la section 2, nous présentons les grandes lignes de l'algorithme d'analyse mis en oeuvre par DYALOG-SR, ainsi que, dans la section 3, les adaptations apportées pour le traitement de treillis de mots. Nous rappelons quelques résultats obtenus par DYALOG-SR pendant la campagne SPRML. Partant de ce point de départ, les expériences menées sur le français sont ensuite décrites en section 4, en précisant certaines modifications supplémentaires apportées à DYALOG-SR pour améliorer ses performances et son efficacité. Enfin, en section 5, nous présentons et discutons les résultats obtenus pour ces expériences sur le corpus FTB, ainsi que sur le corpus hétérogène SEQUOIA.

2 Un algorithme d'analyse par faisceau

DYALOG-SR met en oeuvre une stratégie d'analyse par transitions utilisant le système déductif *arc-standard* de la figure 1(a) sur des configurations formées d'une position j dans la chaîne d'entrée et d'une pile S dont les éléments sont des arbres (partiels) de dépendances. À l'étape m, soit une transition d'empilement (*shift*) empile le prochain mot de la chaîne sur le sommet de la pile, soit une transition de réduction établit une dépendance de label l entre le sommet r_0 de l'arbre s_o comme gouverneur et le sommet r_1 de l'arbre s_1 comme gouverné (réduction $re_{_l\curvearrowleft}$) ou le contraire (réduction $re_{\curvearrowright_l}$). Les configurations sont complétées par un score c mis à jour lors de l'application des transitions par consultation d'un modèle statistique donnant leur coût élémentaire (ξ, λ et ρ). La stratégie *arc-standard* ne permet que de construire des arbres de dépendances projectifs (sans croisement de dépendances) et est proche de la stratégie *arc-eager* [1] généralement mise en oeuvre dans MALT pour les cas projectifs.

entrée : $w_0 \dots w_{n-1}$

axiome $0{:}\langle 0, \epsilon\rangle{:}0$

$$\text{shift}\frac{m{:}\langle j, S\rangle{:}c}{m+1{:}\langle j+1, S|w_j\rangle{:}c+\xi}$$

$$re_{_l\curvearrowleft}\frac{m{:}\langle j, S|s_1|s_0\rangle{:}c}{m+1{:}\langle j, S|s_1\ {}_l\!\curvearrowleft\! s_0\rangle{:}c+\lambda}$$

$$re_{\curvearrowright_l}\frac{m{:}\langle j, S|s_1|s_0\rangle{:}c}{m+1{:}\langle j, S|s_1\ \curvearrowright_l\ s_0\rangle{:}c+\rho}$$

but $2n-1{:}\langle n, s_0\rangle{:}c$

(a) sur les configurations

$$\text{shift}\frac{I = m{:}\langle j, s_0, s_1\rangle{:}(c, \iota)}{\begin{array}{c}J = m+1{:}\langle j+1, w_j, s_0\rangle{:}(c+\xi, \xi)\\ \text{tail}(J) \mathrel{+}= I\\ \text{back}(J) \mathrel{+}= (\text{shift}, I, \text{nil}, c+\xi)\end{array}}$$

$$re_{_l\curvearrowleft}\frac{\begin{array}{c}I = m{:}\langle j, s_0, s_1\rangle{:}(c, \iota)\\ I_t = \_{:}\langle \_, \_, s_2\rangle{:}(c', \iota') \in \text{tail}(I)\end{array}}{\begin{array}{c}J = m+1{:}\langle j, s_1\ {}_l\!\curvearrowleft\! s_0, s_2\rangle{:}(c'+\delta, \iota'+\delta)\\ \delta = \iota + \lambda\\ \text{tail}(J) \cup\!= \text{tail}(I_t)\\ \text{back}(J) \mathrel{+}= ({}_l\!\curvearrowleft, I, I_t, c'+\delta)\end{array}}$$

(b) sur les items en programmation dynamique (fragment)

FIGURE 1 – Systèmes déductifs pour la stratégie *Arc-standard*
les _ dans les items dénotent des champs dont les valeurs ne nécessitent pas d'être consultées

Néanmoins, là où MALT explore l'espace de recherche de manière déterministe et gloutonne en prenant une suite de décisions locales, DYALOG-SR utilise une approche non-déterministe en maintenant, pour chaque étape m, un faisceau de possibilités de largeur k. Pour cela, suivant (Huang & Sagae, 2010), il met en oeuvre des principes de programmation dynamique en identifiant des ensembles de configurations se comportant de manière équivalente par rapport aux transitions et représentables sous forme d'*items*. Plusieurs choix sont possibles pour définir ces items, et dans notre cas, la composante principale d'un item I est fournie par $m{:}\langle j, s_0, s_1\rangle{:}(c, \iota)$ où j dénote la position courante dans la chaîne d'entrée, s_0 et s_1 les 2 premiers éléments de pile (plus précisément des approximations sur les noeuds racines et leurs dépendances immédiates), c le coût préfixe maximum menant à I, et ι le coût *intérieur* maximum depuis l'empilement d'un élément au dessus de s_1, De plus, suivant (Goldberg *et al.*, 2013), nous maintenons pour chaque item I un ensemble d'items queue (*tail*) I_t nécessaires lors des réductions pour retrouver le bas de la pile. Enfin, des pointeurs arrière (typés) sont associés à chaque item permettant de reconstruire une dérivation et un arbre de dépendance à partir d'un item final. Par ailleurs, le système déductif est adapté pour les items comme illustré par la figure 1(b).

L'utilisation de la programmation dynamique signifie en pratique que, même si seuls les k meilleurs items sont conservés à chaque étape m, nettement plus de k configurations sont accessibles au travers des pointeurs arrières. Par contre, à chaque étape, seule l'information présente dans un item est accessible, soit bien moins que dans une configuration complète.

La phase d'apprentissage s'appuie sur les transitions fournies par un oracle et utilise un perceptron moyennée pour mettre

1. Cependant, la stratégie *arc-eager* tend à créer les dépendances au plus tôt, au contraire de *arc-standard*.

à jour le modèle statistique (Daume, 2006), en suivant une stratégie agressive de mise à jour au plus tôt (*early strategy*, (Huang *et al.*, 2012)) dès que l'item oracle $\mathcal{O}_m$ obtenu à l'étape m par application des m premières transitions données par l'oracle se trouve en dehors du faisceau d'items.

Le choix des transitions à appliquer s'appuie sur un jeu assez classique de traits, comprenant en particulier des traits lexicaux liés aux champs du format tabulaire CONLL comme `lex`, `lemma`, `cat` (CPOS), `fullcat` (POS), `mstag` (traits morphosyntaxique FEATS). Ces traits sont déclinés pour le prochain mot la_1 dans la chaîne, pour les deux mots suivants de lecture avant la_2 et la_3 (*lookahead*), et (quand présents) pour les racines r_0 et r_1 des arbres s_0 et s_1 ainsi que pour leurs fils extremum à gauche et à droite. Nous avons aussi des traits liés aux dépendances comme les labels, les valences et domaines, ainsi que les distances discrétisées entre r_0, r_1 et la_1. Les traits ont en général des valeurs atomiques (symboliques ou numériques) mais acceptent également des listes de valeurs, par exemple pour décomposer le trait `mstag` en sous-traits (`gender`, `number`, ...). Ces traits peuvent être combinées sous forme de séquences (*motifs*) pour prendre en compte des corrélations multi-facteurs.

La table 1(a) rappelle quelques résultats obtenus par DYALOG-SR sur le français lors de la campagne SPMRL 2013. Quatre configurations étaient proposées aux participants, avec l'apprentissage réalisé sur un corpus de 14K phrases (`full`) ou un corpus de 5K phrases (`5k`), et avec en entrée un étiquetage de référence (`gold`) ou prédit (`pred`). Le schéma d'annotation utilisé était une variante plus riche du schéma FTB, avec en particulier une relation `dep_cpd` servant à relier les composants d'une forme composée (comme par exemple les composants du terme *motion de censure*). En mode `pred`, la détection de ces relations `dep_cpd` était une des principales difficultés de la tâche.

Dans la table 1(a), les résultats de DYALOG-SR sur la partie `test` sont comparés à ceux du meilleur système, à un système baseline de type MALT, et à la moyenne des résultats fournis par les participants. Les scores sont exprimés en *Labeled Attachement Score* (LAS), en prenant en compte la ponctuation. Enfin, la valeur pour b indique la largeur optimale du faisceau calculée sur la partie `dev`.

On observe que DYALOG-SR reste relativement loin du meilleur système pour toutes les configurations, mais se place néanmoins systématiquement dans la moyenne haute des participants, un résultat somme toute honorable pour un système développé et configuré en environ un mois. Sur l'ensemble des 9 langues, DYALOG-SR se classe second [2], juste devant une version de MALT optimisée sur son jeu de traits grâce à l'emploi de MALTOPTIMIZER (Ballesteros & Nivre, 2012). Enfin, la figure 4(a) montre l'impact des faisceaux dans l'amélioration significative des performances pour toutes les langues, à l'exception encore mystérieuse du coréen.

	DYALOG-SR		autres systèmes		
configuration	test	beam	meilleur	baseline	moyenne
gold/full	87,69	8	90,29	79,86	85,99
gold/5k	85,66	8	88,73	78,16	84,49
pred/full	82,06	8	85,86	77,98	81,03
pred/5k	80,11	4	83,60	76,54	79,31

(a) sur le français (métrique LAS)

	DYALOG-SR	
	chaîne	treillis
baseline beam6	87.34	76.35
modifié + `length` b6		83.47
modifié + `length` b16		86.75
modifié - `length` b16		86.50

(b) sur l'hébreux (métrique TED)

TABLE 1 – Résultats SPMRL 2013 pour le français et l'hébreu

3 Traiter des treillis de mots

Une des particularité de DYALOG-SR réside dans sa capacité à prendre en entrée des treillis de mots. Un tel treillis est un graphe dirigé acyclique (DAG) dont les arcs sont décorés par des mots et dont les noeuds dénotent des positions dans la phrase, comme illustré par la figure 2 pour une phrase en hébreu. Ces treillis sont classiquement utilisés pour représenter des ambiguïtés au niveau des mots mais aussi des ambiguïtés de segmentation et une sous-tâche de SPMRL était dédiée au traitement de treillis, en particulier pour l'hébreu.

Si le traitement de treillis de mots est relativement classique pour les approches d'analyse syntaxique par chartes [3], ce n'est pas le cas pour les approches statistiques et en particulier pour les approches par transitions. En effet, les chemins

2. L'ensemble des résultats sont disponibles dans (Seddah *et al.*, 2013; Villemonte De La Clergerie, 2013a)
3. et est formalisé en tant qu'intersection du langage engendré par une grammaire avec le langage régulier engendré par le treillis

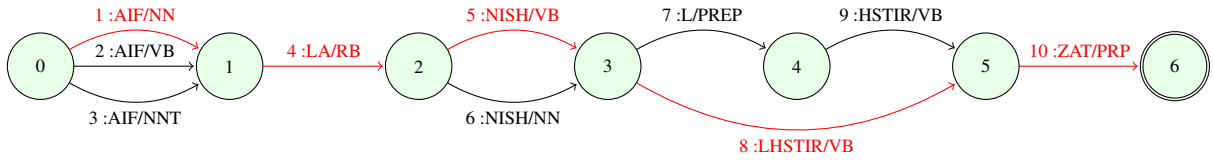


FIGURE 2 – Un treillis ambiguë (avec, en rouge, le chemin de segmentation référence *AIF LA NISH LHSTIR ZAT*)

possibles dans un treillis peuvent être de longueurs différentes, rendant difficile la comparaison des scores associés aux configurations (et aux items). Une première solution à ce problème consiste à normaliser les scores en fonction de la longueur des chemins, mais les expériences menées ne se sont pas révélées très concluantes, même si un gain de quelques points a été obtenu.

Finalement, un trait `length` a été ajouté à chaque mot, dénotant la longueur de l'arc sous-jacent dans le treillis, laissant au mécanisme d'apprentissage le soin de déterminer les meilleurs poids relatifs aux diverses longueurs (en conjonction avec les autres traits).

Une autre difficulté des treillis de mots est lié à l'utilisation des mots en lecture avant. Le choix de ces mots est maintenant non-déterministe et doit rester cohérent avec un chemin valide dans le treillis. En pratique, il devient nécessaire d'étendre les configurations pour y ajouter explicitement le chemin choisi de lecture avant, comme formalisé par le système déductif de la figure 3. Les trois composants $\mathrm{la}_1, \mathrm{la}_2, \mathrm{la}_3$ dénotent des identifiants d'arcs dans le treillis qui forment un début de chemin valide à partir de la position j. La transition `shift` nécessite de choisir un nouvel identifiant la_4 prolongeant le chemin, suite à la consommation de la_1. Les items et transitions sur les items ont similairement été révisés. Bien entendu, l'ajout des identifiants peut grandement multiplier le nombre possible d'items, en gros par le nombre de chemins possibles (de taille 3) dans le treillis, d'où l'importance d'utiliser des faisceaux plus larges.

$$re_{l\curvearrowleft}\frac{m{:}\langle j, S|s_1|s_0, \mathrm{la}_1, \mathrm{la}_2, \mathrm{la}_3\rangle{:}c}{m+1{:}\langle j, S|s_1\ {}_l\!\curvearrowleft s_0, \mathrm{la}_1, \mathrm{la}_2, \mathrm{la}_3\rangle{:}c+\lambda}$$

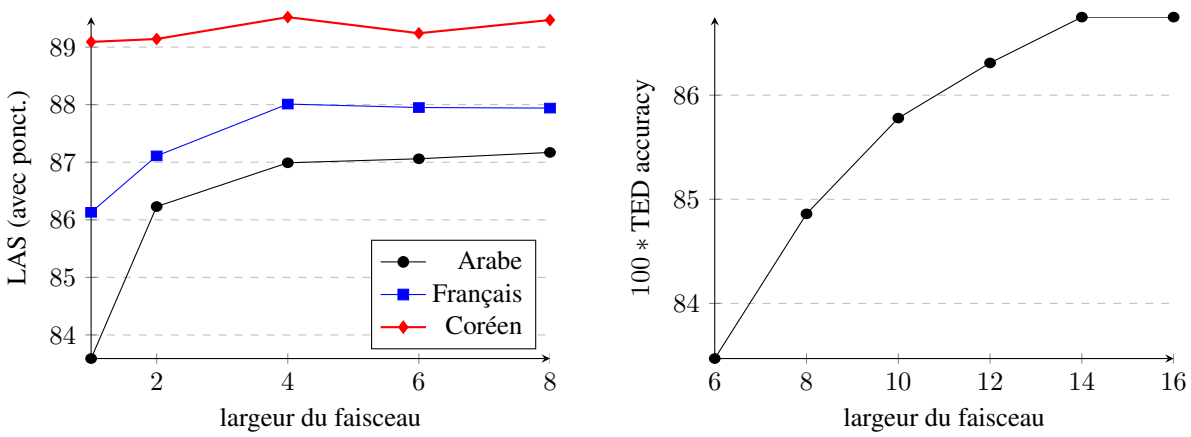
$$re_{\curvearrowright l}\frac{m{:}\langle j, S|s_1|s_0, \mathrm{la}_1, \mathrm{la}_2, \mathrm{la}_3\rangle{:}c}{m+1{:}\langle j, S|s_1\ \curvearrowright_l\ s_0, \mathrm{la}_1, \mathrm{la}_2, \mathrm{la}_3\rangle{:}c+\rho}$$

$$\mathrm{shift}\frac{m{:}\langle j, S, \mathrm{la}_1, \mathrm{la}_2, \mathrm{la}_3\rangle{:}c}{m+1{:}\langle k, S|\mathrm{la}_1, \mathrm{la}_2, \mathrm{la}_3, \mathrm{la}_4\rangle{:}c+\xi}$$

FIGURE 3 – Version révisée de la stratégie arc-standard, pour les treillis

Enfin, lors de la phase d'apprentissage, il est nécessaire de compléter l'oracle avec des informations sur le bon chemin à suivre dans le treillis de manière à pouvoir choisir l'arc suivant la_4 lors des transitions `shift`. En pratique, ce chemin est obtenu par alignement du treillis avec la segmentation de référence.

La table 1(b) donne les résultats obtenus pour l'hébreu, d'abord avec DYALOG-SR non modifié sur des chaînes (non ambiguës) et sur des treillis (ambiguës), puis avec DYALOG-SR modifié (et le trait `length`). Les scores utilisent la métrique TED (Tsarfaty *et al.*, 2011), adaptée à la mise en correspondance d'analyses ne s'appuyant pas sur une même segmentation. La version non modifiée est très mauvaise sur les treillis avec une chute de 11 points mais la version modifiée, pour un faisceau de taille 16, arrive quasiment à combler son retard, avec une perte de seulement 0,6 point. La figure 4(b) montre que des faisceaux plus larges semblent effectivement nécessaires pour compenser l'ambiguité des treillis (avec en moyenne 2,76 arcs par token), mais même avec un faisceau de taille 6, on observe des gains importants par rapport à la version non modifiée. Enfin, de manière surprenante, le trait `length`, bien qu'utile, ne semble pas essentiel et n'assure en définitive qu'un gain de 0,25, peut-être parce qu'il est en partie redondant avec les traits lexicaux.



(a) chaînes (b) treillis hébreux

FIGURE 4 – Influence des faisceaux

4 Trois expériences sur le français

La campagne SPMRL étant maintenant close, il est intéressant de prendre plus de temps et de recul pour explorer les capacités de DYALOG-SR, en particulier sur le français, pour lequel nous disposons d'outils et de ressources supplémentaires pouvant être utiles.

4.1 DYALOG-SR avec étiquetage préalable par MELT

La première expérience a surtout pour vocation de calibrer DYALOG-SR par rapport aux résultats déjà publiés sur le français, en particulier sur la version en dépendances du FTB (Candito *et al.*, 2010b; Urieli & Tanguy, 2013). Nous avons utilisé une version du FTB étiquetée et lemmatisée avec MELT [4] (Denis & Sagot, 2009) en mode validation croisée 10 fois avec une f-mesure de 97,88% sur la partie test du FTB, comparable au taux de 97,81% mentionné dans (Urieli & Tanguy, 2013). Les traits morpho-syntaxiques sont fournis par le lexique LEFFF (Sagot *et al.*, 2006), que nous utilisons également pour fournir des informations de sous-catégorisation pour les verbes. Nous incluons de plus des traits de *clustering* appris sur un large corpus par application de l'algorithme de Brown (Liang, 2005). Suivant (Urieli & Tanguy, 2013), nous avons également ajouté quelques traits liés à la présence (ou non) de ponctuations entre s_0 et le premier mot à lire la_1.

Enfin, nous avons réalisé que les règles de déduction sur les items (fig. 1(b)) permettent en fait d'accéder à l'élément de pile s_2 lors des réductions (au travers des items de queue I_t) et qu'il est donc possible d'utiliser des traits liés à s_2 lors des réductions. Par contre, pour les transitions `shift`, nous ne cherchons pas à retrouver s_2 et les traits associés.

Ces traits supplémentaires ont été également utilisés pour les deux autres expériences.

Tout dernièrement, nous avons également cherché à rendre encore plus agressive la stratégie de mise à jour du modèle lors de l'apprentissage. En particulier, en sus des règles existantes, le modèle est maintenant mis à jour dès que le coût de la dernière action conduisant à l'item oracle $\mathcal{O}_m$ se trouve en dessous d'une certaine marge avec le coût de la dernière action conduisant au meilleur item, ce même si $\mathcal{O}_m$ se trouve dans le faisceau. La valeur de la marge est automatiquement ajustée en fonction du coût moyen des transitions. Enfin, lors d'une mise à jour, le perceptron normalement incrémente ou décrémente les poids de 1 pour les traits concernés, mais cet incrément est maintenant porté à 2 en cas de détection de pertes de dépendances, c'est à dire quand le choix de la transition sur l'item à pénaliser conduit à perdre au moins une dépendance prédite par l'oracle.

De nouveaux tests réalisés avec la nouvelle version de DYALOG-SR sur les données SPRML du français (pred/full) ont montré une évolution des performances sur la partie `dev` de 82,88 à 83,51.

4. Ces données nous ont été gentiment préparées et fournies par Benoît Sagot.

4.2 DYALOG-SR sur des treillis produits par SXPIPE

Le fait que DYALOG-SR puisse prendre des treillis de mots en entrée nous a incité à le tester sur les treillis fournis par SXPIPE pour le français (Sagot & Boullier, 2008), avec des information lexicales fournies par LEFFF. En pratique, SXPIPE est configuré pour respecter le découpage en tokens fourni par le FTB mais peut reconnaître des lexèmes couvrant plusieurs tokens FTB (par exemple pour des entités nommées) ou, au contraire, avoir plusieurs lexèmes couvrant un même token FTB. Pour pouvoir à terme réaligner les treillis SXPIPE avec les données du FTB, nous leurs appliquons des transformations supplémentaires, comme illustré par la table 2.

token(s) FTB	**lexème(s) SXPIPE**	**après transformation**
10 avril	_DATE_artf/`nc`	10/`nc` : `cmpd` (part=1) avril/`nc` : `cmpd` (part=2)
en_surface	en/`prep` surface/`nc`	en_surface/`P+V`
avec_un_bel_ensemble	avec/`prep` un/`det` bel/`adj` ensemble/`nc`	avec_un_bel_ensemble/`aggl`

TABLE 2 – Exemples de transformations pour SXPIPE

Ainsi, un lexème SXPIPE multi-tokens de catégorie `X` est découpé en morceaux pour chaque token, avec des catégories de la forme `X` : `cmpd`. Un trait `part=i` est aussi ajouté aux traits morpho-syntaxiques pour chaque composant i. Dans l'autre sens, les lexèmes SXPIPE couvrant un même token FTB sont fusionnés. Ainsi la séquence `de/prep les/det` associée au token `des` est fusionnée en `des/prep+det`. Dans certains cas, ce processus de fusion peut amener à combiner un composant `X` : `cmpd` avec d'autres lexèmes.

Le processus de fusion crée un nombre très important de catégories complexes $X_1+\cdots+X_n$ (autour de 10 000 catégories), ce qui nous a amené à en rejeter certaines (hautement improbables) et surtout à remplacer toutes les séquences avec $n > 2$ par la catégorie générique `aggl`. Au final, nous obtenons 188 catégories, ce qui représente encore un nombre très conséquent de catégories. Par ailleurs, la plupart des catégories simples (correspondant à celles du LEFFF) ont été converties vers les catégories simples du French TreeBank (`N`, `V`, `A`, `ADV`, `PRO`, `C`, ...). Cette conversion facilite en particulier l'alignement avec les données FTB pour retrouver le chemin de référence dans les treillis lors de la phase d'apprentissage. Les lemmes et traits morpho-syntaxiques sont également exploités pour cet alignement. La table 3 fournit quelques statistiques sur les treillis obtenus pour la partie test du FTB, avec ainsi 2 arcs en moyenne par token FTB.

#tokens	**#arcs**	**arcs/token**	**#catégories**	**dont X+Y**	**#(arcs cmpd)**	**#(arcs X+Y)**	**#(arcs aggl)**
278 083	560 176	2,01	188	131	30 980 (5,5%)	13 174 (2,3%)	2 318 (0,4%)

TABLE 3 – Quelques statistiques sur les treillis SXPIPE pour FTB test

Nos premiers tests ont permis de mettre en évidence des faiblesses dans l'algorithme initial de DYALOG-SR qui n'avaient pas été perçues pour l'hébreu. En effet, pour un faisceau de taille 1, les scores se situaient très bas, entre 40% et 45%, sur les corpus de développement et de test. Une analyse des résultats a montré que le choix de la bonne étiquette syntaxique posait souvent problème. Nous avons alors réalisé que la transition `shift` (fig. 3) ne prenait pas en compte le prochain mot en avant la_4 (le plus à droite) pour évaluer son score. Comme la_4 est ensuite intégré dans le nouvel item et ne peut plus être remis en cause, cela conduisait à un mauvais étiquetage, et en cascade à de mauvaises dépendances. Nous avons donc modifié DYALOG-SR pour prendre en compte des traits relatifs à la_4 lors des transitions `shift` et avons également ajouté quelques informations relatives au token suivant la_4 comme l'ensemble des catégories possibles.

Les expériences menées sur les treillis pour l'hébreu ont montré l'importance de faisceaux dont la largeur est en relation avec le niveau d'ambiguité de ceux-ci. Plutôt que d'augmenter globalement cette largeur, nous avons testé un mécanisme d'adaptation locale de la largeur du faisceau en fonction du niveau local d'ambiguïté. Empiriquement, pour l'étape m, nous appliquons un coefficient multiplicateur donné par $b(i) = \max(1, \log(|\mathcal{P}_{i,3}|))$ pour les positions i entre $m/2 \pm 2$ avec $\mathcal{P}_{i,3}$ dénotant l'ensemble des chemins de longueur 3 partant de i. Cela signifie que même pour un faisceau globalement de largeur 1, nous pouvons localement avoir une largeur > 1. En pratique, sur le FTB, $b(i)$ se situe entre 1 et 5.

Cependant, le coût en temps sur les treillis reste relativement élevé, en particulier à cause du mombre d'items créés et finalement rejetés car étant hors du faisceau. L'article (Choi & McCallum, 2013) nous a fourni une piste pour contrer ce phénomène : il suggère en effet que de bons résultats peuvent étre obtenus en ne considérant que les 2 meilleures configurations par étape, en se limitant de plus aux étapes à faible confiance, c'est à dire les étapes où les coûts des meilleures actions se situent dans une faible marge. Cela nous a conduit, pour DYALOG-SR, à ne considérer que les k meilleurs items dérivables par réduction à partir d'un item I, à condition que leurs coûts soient dans une certaine marge

m. Par défaut, nous prenons $k = 3$ (au lieu de 2 dans le cas de (Choi & McCallum, 2013)) et $m = 200$ (pour des coûts élémentaires de l'ordre de ± 1000).

4.3 DYALOG-SR guidé par FRMG

FRMG est une grammaire d'arbres adjoints (TAG) à large couverture du français, dérivée d'une description grammaticale de haut-niveau sous forme d'une méta-grammaire (de La Clergerie, 2005b). [5] Un analyseur par charte, compilé à partir de la grammaire, permet de retourner l'ensemble des analyses complètes possibles pour une phrase, sous forme d'une forêt partagée de dérivations. En cas d'échec de l'analyse, l'analyseur bascule en mode *robuste* et fournit des séquences d'analyses partielles, de manière à couvrir la phrase. La forêt de dérivation est convertie en forêt de dépendances, et est ensuite désambiguisée à l'aide d'un ensemble de règles heuristiques portant sur les dépendances. Enfin, l'arbre de dépendances ainsi obtenu peut être converti suivant divers schémas d'annotation syntaxique, dont celui utilisé pour les campagnes EASy et Passage ainsi que celui utilisé pour la version dépendance du FTB (schéma FTB). Récemment, des techniques d'apprentissage partiellement supervisé ont permis de fortement améliorer la phase de désambiguisation de FRMG (Villemonte De La Clergerie, 2013b), pour amener ses performances au niveau de celles d'analyseurs statistiques.

Disposant de deux analyseurs syntaxiques de bonne qualité pour le français, il était intéressant de les combiner. Dans le cas présent, nous avons choisi d'utiliser les résultats de FRMG pour guider les décisions de DYALOG-SR. Plus précisément, nous avons ajouté des traits statiques `frmg_label` et `frmg_delta`, indiquant respectivement le label de la dépendance entrante sur chaque mot, et la distance (positive ou négative) à son gouverneur. Les résultats obtenus avec ces traits préliminaires étant déjà extrêmement encourageants, nous avons alors recherché des traits plus proches d'un vrai guidage, liés aux configurations et pas seulement aux tokens.

Ainsi, lorsqu'un item I avec les 2 éléments de piles s_0 et s_1 de racines r_0 et r_1 est compatible avec une réduction donnant une dépendance de label l prédite par FRMG entre r_0 et r_1, un trait dynamique de guidage `reduce_right` ou `reduce_left` est ajouté, lors de l'exécution. Comme les réductions droites peuvent être en compétition avec une transition `shift`, un trait de guidage `shift` est également émis si le premier mot la_1 précède ou égale le descendant de r_0 le plus à droite pour FRMG. Enfin, le label l de la dépendance est ajouté au trait de guidage.

5 Résultats et discussions

Les trois expériences précédemment décrites ont été conduites sur la version en dépendances du FTB (Candito *et al.*, 2010a), en utilisant classiquement la partie `train` (9881 phrases) pour l'entraînement, la partie `dev` (1235 phrases) pour les mises au point et la partie `test` (1235 phrases) pour l'évaluation.

La table 5(a) fournit les performances en LAS (sans prise en compte de la ponctuation) pour les diverses versions de DYALOG-SR, pour FRMG, et pour divers analyseurs statistiques. Pour FRMG, nous fournissons les performances pour la version de base et pour une version `+tuning` après amélioration de la phase de désambiguisation par apprentissage sur la partie `train` du FTB. C'est cette version `+tuning` qui est utilisée pour l'expérience de couplage. Pour les analyseurs statistiques, nous reprenons comme baseline les résultats publiés dans (Candito *et al.*, 2010b) pour les systèmes BERKELEY (constituance + conversion en dépendances), MALT et MST (*Maximum Spanning Tree*).

Nous avons ensuite les résultats plus récents du système TALISMANE (Urieli & Tanguy, 2013), qui partage un certain nombre de points communs avec DYALOG-SR, comme l'emploi d'un algorithme par transition, l'emploi de faisceaux et l'emploi de LEFFF. TALISMANE intègre aussi, en plus des traits statistiques, des règles/contraintes linguistiques censées bloquer certaines configurations impossibles, ce qui a, en partie, motivée le couplage de DYALOG-SR avec une source d'information linguistique comme FRMG.

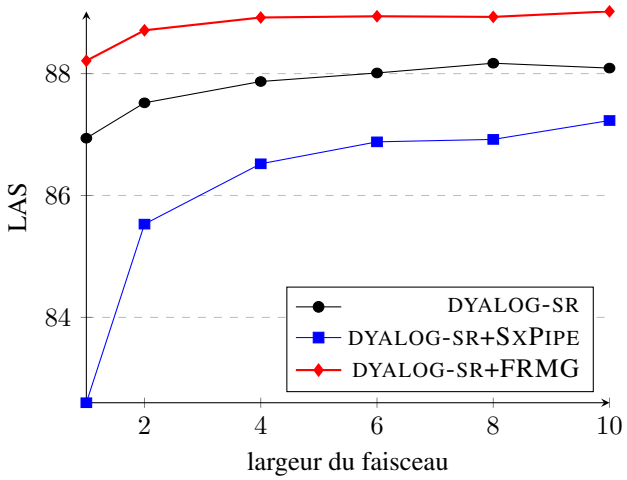
Enfin, nous mentionnons les résultats publiés pour le français dans (Le Roux *et al.*, 2012) qui s'appuient sur l'application d'un modèle discriminatif pour le ré-ordonnancement des n meilleurs analyses produites par une grammaire PCFG (avec un modèle génératif) et transformées en dépendances. La méthode donne de très bons résultats et présente quelques similarités avec l'idée de coupler plusieurs modèles statistiques. Pour information, nous donnons aussi les résultats obtenus par MATE (Bohnet, 2010), une version améliorée de MST exploitant des informations fournies par MELT.

Avec un étiquetage préalable par MELT, DYALOG-SR obtient de bons résultats, similaire à ceux de MALT pour un faisceau

5. Accessible en ligne sur `http://alpage.inria.fr/frmgwiki`.

système	dev	test
FRMG init	80,85	82,08
FRMG +tuning	86,20	87,49
BKY	86,50	86,80
MALT	86,90	87,30
MST	87,50	88,20
TALISMANE (b=1)	86,80	87,20
TALISMANE (b=20)	88,10	88,50
Le Roux (2012)	–	89,20
MATE+MELT	–	89,20
DYALOG-SR (b=1 i=5)	86,94	87,71
DYALOG-SR (b=8 i=10)	88,17	89,01
DYALOG-SR treillis (b=1 i=8)	82,60	83,68
DYALOG-SR treillis (b=10 i=9)	87,23	87,90
DYALOG-SR treillis (b=12 i=7)	87,15	88,15
DYALOG-SR+FRMG (b=1 i=6)	88,21	89,38
DYALOG-SR+FRMG (b=10 i=7)	89,02	90,25

(a) comparaison des systèmes



(b) impact des faisceaux (sur FTB `dev`)

FIGURE 5 – Performances sur le FTB (métrique LAS, sans ponctuations)

de taille 1. Les meilleurs résultats, obtenus avec un faisceau de taille 8 (et itération 10), dépassent ceux de la 1ère génération d'analyseurs (BKY, MALT et MST) et même ceux plus récemment obtenus par TALISMANE. Ils s'approchent des scores atteints par (Le Roux *et al.*, 2012). Il est intéressant de noter que des choix locaux en partie compensés par un faisceau assez large rivalisent avec des choix plus globaux, comme mis en oeuvre par MST ou MATE ou par réordonnancement dans (Le Roux *et al.*, 2012).

Les résultats obtenus par DYALOG-SR sur les treillis SXPIPE, sans étiquetage préalable et avec réalignement, sont finalement assez proches de ceux obtenus avec étiquetage, avec une perte d'un peu plus d'un point, malgré un jeu très important de 188 catégories syntaxiques. Ils dépendent cependant fortement de l'utilisation des faisceaux. La figure 5(b) montre en effet que les performances croissent de manière importante avec la largeur des faisceaux, avec plus de quatre points de gains, mais que grâce à l'adaptation locale des faisceaux, il n'est pas cependant nécessaire d'utiliser des faisceaux aussi larges que pour l'hébreu. Il est probable qu'il existe encore une marge de progression, en intégrant de nouveaux traits plus spécifiquement dédiés au choix du prochain lexème dans le treillis et en gérant mieux les processus de fusion des lexèmes SXPIPE. Comme les catégories syntaxiques ne sont pas celles du FTB, il est difficile de fournir une évaluation précise de l'étiquetage finalement obtenu. Néanmoins, en projetant sur les 15 catégories simples du FTB, on obtient une f-mesure de 95,35% sur FTB test (et $b = 10, i = 9$) alors qu'elle était autour de 70% avant les modifications apportées à la gestion de la_4 lors des transitions `shift`.

Enfin, les résultats les plus intéressants sont ceux fournis par le couplage de DYALOG-SR avec FRMG. On observe que les scores obtenus sont bien meilleurs que ceux obtenus par FRMG et DYALOG-SR pris isolément, et qu'à notre connaissance, ils sont les meilleurs à ce jour sur le FTB. Les deux analyseurs semblent donc complémentaires, ce qui conforte l'intérêt d'injecter des connaissances linguistiques dans un analyseur statistique, ici au travers des traits de guidage.

Pour confirmer cette complémentarité, nous avons examiné plus précisément le comportement de FRMG, DYALOG-SR, et DYALOG-SR+FRMG pour certaines relations de dépendance, comme illustré par la figure 6. On observe ainsi que DYALOG-SR+FRMG dépasse presque systématiquement les 2 autres analyseurs, en rappel et en précision. On note que certaines dépendances difficiles pour les analyseurs statistiques, comme `coord` et `mod_rel`, bénéficient pleinement des informations fournies par un analyseur linguistique comme FRMG, et que DYALOG-SR semble même capable de généraliser les informations de FRMG pour le dépasser. Dans l'autre sens, la faiblesse de FRMG sur la relation `mod` et, en rappel, sur les relations pour les arguments verbaux propositionnels (`a_obj`, `de_obj`, `p_obj`) ne perturbe pas le couplage. Enfin, le mode robuste, utilisé par FRMG quand il ne trouve pas d'analyse complète, introduit de fausses racines, ce qui fait chuter sa précision sur le label `root`. Mais le couplage résout totalement ce problème.

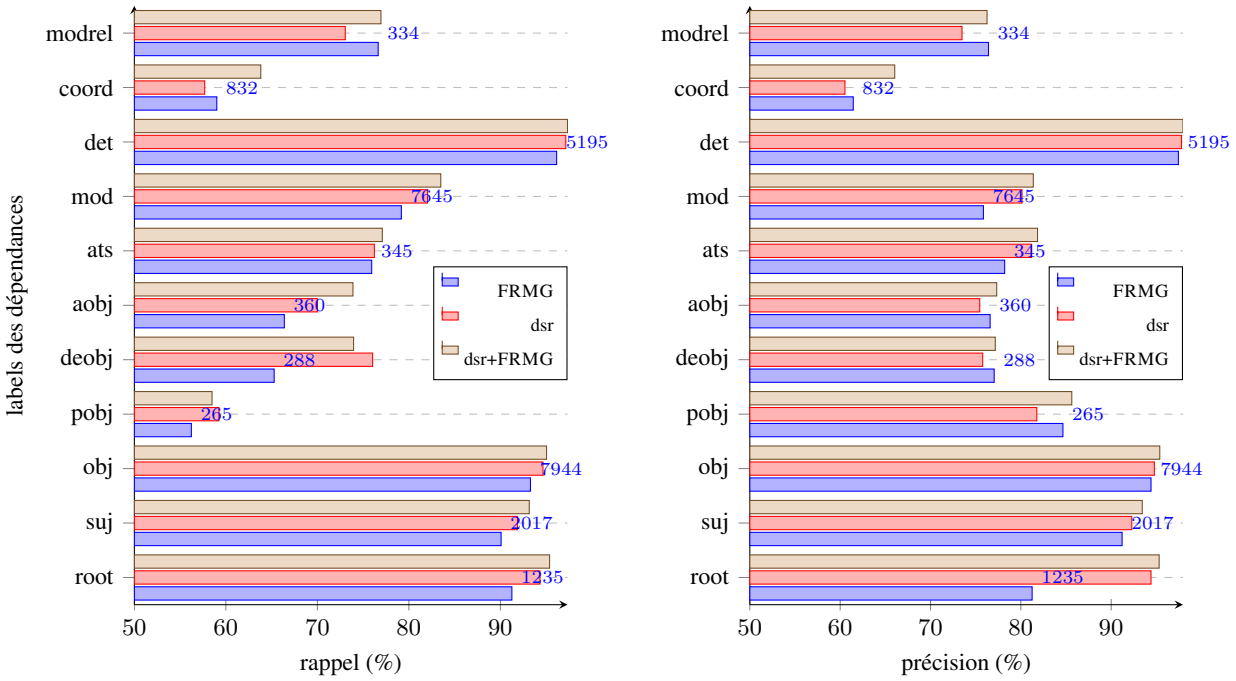


FIGURE 6 – Comparaison des systèmes sur les labels des dépendances
le volume de dépendances concernées est fourni en regard des barres

Toujours pour étudier les différences entre les analyseurs, nous les avons comparé selon divers critères *topologiques*, comme la distance entre gouverneur et gouverné (fig. 7(a)), la profondeur dans l'arbre de dépendance (fig. 7(b)), la taille de la fratrie (fig. 7(c)), et enfin le rang (gauche ou droit) dans la fratrie (fig. 7(d)). Nous notons ainsi que pour les dépendances à longue distance, difficiles pour un analyseur local comme DYALOG-SR, nous obtenons de bien meilleurs résultats par couplage avec FRMG. Pour la profondeur, qui pose problème à FRMG [6], nous notons une nette amélioration de DYALOG-SR+FRMG, en rappel et en précision. Pour le rang, et en particulier pour les rangs négatifs (dépendances gauches) qui posent problème pour DYALOG-SR, on voit que le couplage hérite des meilleures performances de FRMG. Enfin, la trace des faiblesses du mode robuste pour FRMG est bien visible en précision pour la distance nulle, la profondeur nulle et le rang nul, mais ces faiblesses sont totalement compensées pour DYALOG-SR+FRMG.

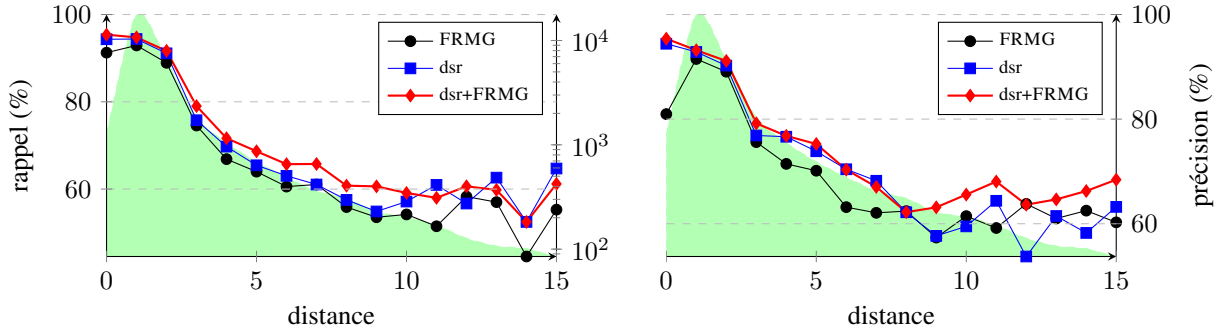
Enfin, nous avons examiné la stabilité des analyseurs sur le corpus SEQUOIA (Candito & Seddah, 2012). Les 3 204 phrases de ce corpus sont aussi annotées suivant le schéma du FTB mais couvrent divers domaines autres que le domaine journalistique du FTB. En particulier, les sous-parties EMEA correspondent à des textes médicaux difficiles à traiter. Pour des analyseurs entraînés sur FTB train, la table 4 montre que les pertes, par rapport à FTB test, sont plus faibles pour FRMG que pour DYALOG-SR, avec même FRMG dépassant DYALOG-SR en performance sur certains sous-corpus (Europar, annodis, et emea-fr-dev). [7] On observe par contre que le couplage préserve partiellement la stabilité fournie par FRMG et permet d'obtenir de bons scores sur SEQUOIA.

Conclusion

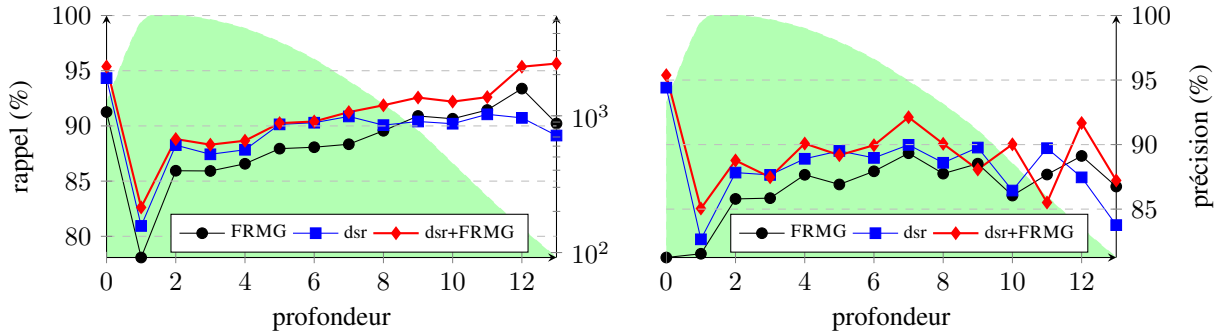
Nous avons présenté trois expériences menées sur le français et s'appuyant sur l'analyseur DYALOG-SR. Celles-ci montrent qu'un analyseur relativement simple par transition est néanmoins susceptible d'obtenir de bons résultats. Sa simplicité permet de facilement et rapidement tester diverses hypothèses et extensions, comme l'adaptation locale des faisceaux dans les treillis, et les expériences menées nous ont déjà permis de faire largement évoluer DYALOG-SR depuis ses débuts dans la campagne SPMRL 2013. Néanmoins, même si les faisceaux ont un impact important sur les performances et permettent

6. à priori parce que le modèle de FRMG n'utilise pas de traits directement ou indirectement liés à la profondeur, au contraire de DYALOG-SR avec les traits liés à s_0, s_1 et s_2.

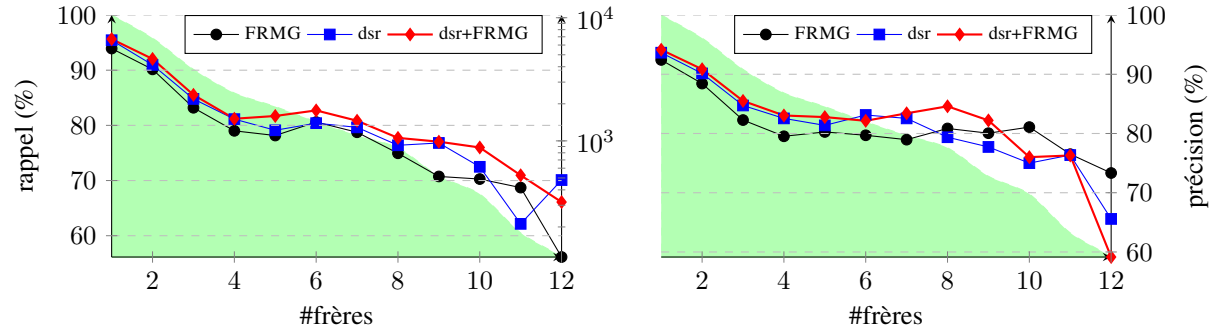
7. Les mauvaises performances de FRMG sur Wikipedia semblent dues à quelques longues phrases provoquant un timeout et une perte complète d'analyses.



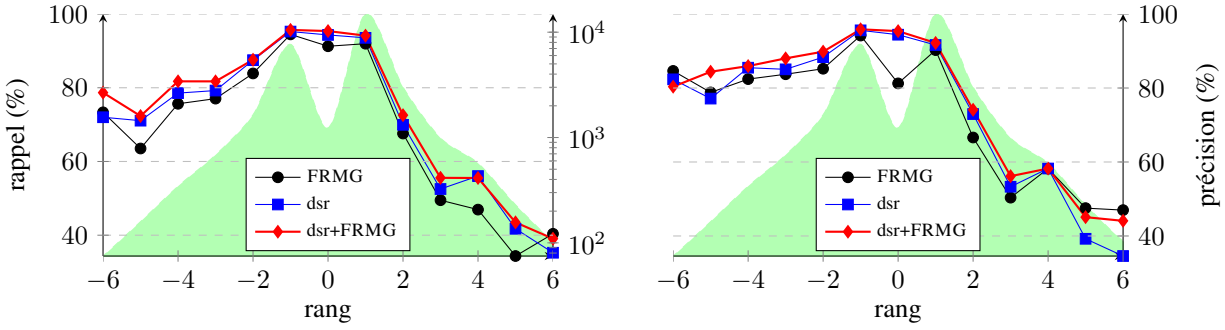
(a) par rapprort à la distance



(b) par rapport à la profondeur



(c) par rapport au nombre de frères



(d) par rapport au rang dans la fratrie

FIGURE 7 – Comparaison des systèmes selon divers critères *topologiques*
les fonds (en vert) indiquent le volume de dépendances concernées (échelle logarithmique à droite)

		FRMG			DYALOG-SR			DYALOG-SR+FRMG		
corpus	#phrases	LAS	Δerr	%Δerr	LAS	Δerr	%Δerr	LAS	Δerr	%Δerr
FTB test	1235	87,49	–	–	89,00	–	–	90,25	–	–
Europar	561	87,97	−0,5	−3,8	87,00	+2,0	+18,2	88,94	+1,3	+13,4
annodis	529	86,11	+1,4	+11,0	85,80	+3,2	+29,1	88,21	+2,0	+20,9
emea-fr-dev	574	85,16	+2,3	+18,6	83,50	+5,5	+50,0	86,26	+4,0	+40,9
emea-fr-test	544	84,67	+2,8	+22,5	85,01	+4,0	+36,3	86,87	+3,4	+34,7
frwiki	996	83,53	+4,0	+31,7	84,39	+4,6	+41,9	86,23	+4,0	+41,2

TABLE 4 – Stabilité des systèmes sur le corpus hétérogène SEQUOIA, après apprentissage sur FTB

en partie de corriger les choix locaux faits un analyseur par transition (comme MALT, TALISMANE ou DYALOG-SR), il paraît difficile à terme de concurrencer des approches plus globales, éventuellement obtenues par couplage de modèles.

En allant dans cette direction, l'expérience de couplage de DYALOG-SR avec FRMG nous a ainsi fourni d'excellents résultats. Elle confirme l'intérêt d'injecter une information plus linguistique dans un analyseur statistique comme DYALOG-SR (et comme tenté aussi dans TALISMANE). Elle confirme également l'intérêt d'exploiter des contraintes de *localité étendue*, telles que fournis (indirectement) par les arbres TAG de FRMG pour mieux guider les choix plus locaux de DYALOG-SR. Il est aussi intéressant de noter que ce couplage ne permet pas seulement d'hériter des meilleures propriétés de chaque système, mais permet en fait de les dépasser.

Le couplage de plusieurs analyseurs statistiques n'est pas nouveau (Sagae & Lavie, 2006), mais celui d'un analyseur statistique avec un analyseur fondé sur une grammaire linguistique est déjà plus original et prometteur (Ovrelid *et al.*, 2009). (Villemonte De La Clergerie, 2013b) a montré qu'il est possible d'améliorer les performances d'un analyseur symbolique en lui permettant d'apprendre à partir d'un corpus annoté comme le FTB, et nous voyons ici que la combinaison d'un analyseur statistique avec un analyseur symbolique est aussi une piste intéressante pour fortement améliorer les performances, ce qui est une incitation à utiliser au mieux les divers analyseurs du français qui existent (et pas forcément les seuls analyseurs statistiques entraînés sur le FTB). Cette approche, qui reste néanmoins assez lourde à mettre en oeuvre, assure aussi une meilleure stabilité pour des corpus hors domaine d'entraînement, comme montré sur le corpus SEQUOIA.

Ce couplage ouvre également la voie à d'autres expériences. Ainsi, l'examen fin des améliorations apportées par le couplage peut sûrement permettre d'identifier et de corriger certaines faiblesses de FRMG, en particulier pour le mode robuste. D'autre part, dans le cas présent, nous avons utilisé les informations de FRMG pour guider DYALOG-SR, mais il est aussi envisageable d'exploiter les résultats de DYALOG-SR pour guider FRMG, d'autant plus que DYALOG-SR peut travailler sur les mêmes treillis SXPIPE servant d'entrée à FRMG. Par ailleurs, les sorties de FRMG sur de gros corpus peuvent être exploitées pour entraîner DYALOG-SR. Enfin l'expérience de guidage que nous avons menée après étiquetage préalable du FTB pourrait aussi être tentée directement sur les treillis SXPIPE, modulo quelques efforts d'alignement pour attacher les informations sur les bons arcs des treillis.

Enfin, DYALOG-SR continue d'évoluer. Dans le cadre de sa participation à la campagne SEMEVAL 2014 (tâche 8), il a été modifié pour produire des graphes de dépendances sémantiques (plutôt que des arbres de dépendances syntaxiques), en s'appuyant sur un jeu plus riche de transitions, permettant de définir une stratégie de type arc-eager (attachement au plus tôt) ainsi qu'une forme faible de non-projectivité. D'autres développements sont prévus pour un traitement plus complet des cas de non-projectivité.

Références

BALLESTEROS M. & NIVRE J. (2012). MaltOptimizer : an optimization tool for MaltParser. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, p. 58–62.

BOHNET B. (2010). Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of COLING*.

CANDITO M., CRABBÉ B. & DENIS P. (2010a). Statistical French dependency parsing : treebank conversion and first results. In *Proceedings of the 7th Language Resources and Evaluation Conference (LREC'10)*, La Valette, Malte.

CANDITO M., NIVRE J., DENIS P. & HENESTROZA ANGUIANO E. (2010b). Benchmarking of statistical dependency parsers for French. In *Proceedings of COLING'2010 (poster session)*, Beijing, China.

CANDITO M. & SEDDAH D. (2012). Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *TALN 2012-19e conférence sur le Traitement Automatique des Langues Naturelles*.

CHOI J. D. & MCCALLUM A. (2013). Transition-based dependency parsing with selectional branching. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, Sofia, Bulgaria*.

DAUME H. C. (2006). *Practical structured learning techniques for natural language processing*. PhD thesis, University of Southern California.

DE LA CLERGERIE É. (2005a). DyALog : a tabular logic programming based environment for NLP. In *Proceedings of 2nd International Workshop on Constraint Solving and Language Processing (CSLP'05)*, Barcelone, Espagne.

DE LA CLERGERIE É. (2005b). From metagrammars to factorized TAG/TIG parsers. In *Proceedings of IWPT'05*, p. 190–191, Vancouver, Canada.

DENIS P. & SAGOT B. (2009). Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art POS tagging with less human effort. In *Proceedings PACLIC 23*, Hong Kong, China.

GOLDBERG Y., ZHAO K. & HUANG L. (2013). Efficient implementation of beam-search incremental parsers. In *Proc. of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, Sophia, Bulgaria.

HUANG L., FAYONG S. & GUO Y. (2012). Structured perceptron with inexact search. In *Proceedings of HLT-NAACL 2012*, p. 142–151.

HUANG L. & SAGAE K. (2010). Dynamic programming for linear-time incremental parsing. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 1077–1086 : Association for Computational Linguistics.

LE ROUX J., FAVRE B., NASR A. & MIRROSHANDEL S. (2012). Generative constituent parsing and discriminative dependency reranking : Experiments on english and french. In *SP-SEM-MRL*.

LIANG P. (2005). Semi-supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology.

NIVRE J. (2003). An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT 03)*, p. 149–160, Nancy, France.

OVRELID L., KUHN J. & SPREYER K. (2009). Improving data-driven dependency parsing using large-scale LFG grammars. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, p. 37–40 : Association for Computational Linguistics.

SAGAE K. & LAVIE A. (2006). Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume : Short Papers*, p. 129–132.

SAGOT B. & BOULLIER P. (2008). SXPIPE 2 : architecture pour le traitement présyntaxique de corpus bruts. *Traitement Automatique des Langues (T.A.L.)*, **49**(2), 155–188.

SAGOT B., CLÉMENT L., DE LA CLERGERIE É. & BOULLIER P. (2006). The Le*fff* 2 syntactic lexicon for French : architecture, acquisition, use. In *Proceedings of the 5th Language Resources and Evaluation Conference (LREC'06)*, Genova, Italie.

SEDDAH D., TSARFATY R., KÜBLER S., CANDITO M., CHOI J., FARKAS R., FOSTER J., GOENAGA I., GOJENOLA K., GOLDBERG Y., GREEN S., HABASH N., KUHLMANN M., MAIER W., NIVRE J., PRZEPIORKOWSKI A., ROTH R., SEEKER W., VERSLEY Y., VINCZE V., WOLIŃSKI M., WRÓBLEWSKA A. & VILLEMONTE DE LA CLERGERIE E. (2013). Overview of the SPMRL 2013 shared task : A cross-framework evaluation of parsing morphologically rich languages. In *Proceedings of the 4th Workshop on Statistical Parsing of Morphologically Rich Languages : Shared Task*, Seattle, WA.

TSARFATY R., NIVRE J. & ANDERSSON E. (2011). Evaluating dependency parsing : Robust and heuristics-free cross-annotation evaluation. In *Proceedings of the 8th International Conference on Empirical Methods in Natural Language Processing (EMNLP 11)*, Edinburgh, UK.

URIELI A. & TANGUY L. (2013). L'apport du faisceau dans l'analyse syntaxique en dépendances par transitions : études de cas avec l'analyseur Talismane. In *Actes de la 20e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2013)*, p. 188–201, Les Sables d'Olonne, France.

VILLEMONTE DE LA CLERGERIE É. (2013a). Exploring beam-based shift-reduce dependency parsing with DyALog : Results from the SPMRL 2013 shared task. In *4th Workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL'2013)*, Seattle, États-Unis.

VILLEMONTE DE LA CLERGERIE É. (2013b). Improving a symbolic parser through partially supervised learning. In *The 13th International Conference on Parsing Technologies (IWPT)*, Nara, Japon.

Principes de modélisation systémique des réseaux lexicaux *

Alain Polguère
ATILF, CNRS & Université de Lorraine, 44 av. de la Libération, BP 30687 , 54063 Nancy Cedex
alain.polguere@univ-lorraine.fr

Résumé. Nous présentons une approche de la construction manuelle des ressources lexicales à large couverture fondée sur le recours à un type particulier de réseau lexical appelé *système lexical*. En nous appuyant sur l'expérience acquise dans le cadre de la construction du Réseau Lexical du Français (RL-fr), nous offrons tout d'abord une caractérisation formelle des systèmes lexicaux en tant que graphes d'unités lexicales de type « petits mondes » principalement organisés à partir du système des fonctions lexicales Sens-Texte. Nous apportons ensuite des arguments pour justifier la pertinence du modèle proposé, tant du point de vue théorique qu'applicatif.

Abstract. We introduce a new approach for manually constructing broad-coverage lexical ressources based on a specific type of lexical network called *lexical system*. Drawing on experience gained from the construction of the French Lexical Network (fr-LN), we begin by formally characterizing lexical systems as "small-world" graphs of lexical units that are primarily organized around the system of Meaning-Text lexical functions. We then give arguments in favor of the proposed model that are both theory- and application-oriented.

Mots-clés : système lexical, base de données lexicale, structure du lexique, réseau lexical, graphe petit monde, proxémie, Lexicologie Explicative et Combinatoire, fonction lexicale, Réseau Lexical du Français (RL-fr).

Keywords: lexical system, lexical database, structure of the lexicon, lexical network, small-world graph, proxemy, Explanatory Combinatorial Lexicology, lexical function, French Lexical Network (fr-LN).

1 Introduction

Cet article rend compte des principes de structuration de la connaissance lexicale mis en pratique dans le cadre de la construction manuelle, par une équipe de lexicographes, du Réseau Lexical du Français, désormais RL-fr. Cette ressource est élaborée suivant le cadre théorique et descriptif général de la Lexicologie Explicative et Combinatoire (Mel'čuk *et al.*, 1995; Mel'čuk, 2006). Elle se distingue cependant radicalement des travaux lexicographiques antérieurement effectués selon cette approche par le fait qu'elle repose sur un modèle non dictionnairique et non textuel appelé *système lexical*.

La première étape de construction du RL-fr, qui doit s'achever fin 2014, s'est effectuée dans le cadre des travaux du projet RELIEF, menés au laboratoire ATILF du CNRS. Plusieurs publications traitent de l'implémentation du RL-fr, de son édition lexicographique et de sa croissance (Lux-Pogodalla & Polguère, 2011; Gader *et al.*, 2012; Polguère & Sikora, 2013). Nous nous intéressons ici spécifiquement au type particulier de modélisation systémique des lexiques qu'il implémente, en nous concentrant sur deux points : nature formelle du système lexical du RL-fr (section 2) et justification de l'approche adoptée pour organiser l'information lexicale à travers l'examen de son potentiel applicatif (section 3).

Les systèmes lexicaux appartiennent à la grande famille des réseaux lexicaux, dont la branche issue de WordNet (Fellbaum, 1998) est de loin la plus connue en Traitement Automatique des Langues. Dans ce contexte, il est sans doute utile de rappeler, avant d'entrer dans le vif du sujet, que l'on peut distinguer deux grands types de réseaux lexicaux.

1. Les réseaux ontologiquement organisés, comme WordNet, structurent l'information lexicale de façon prioritairement hiérarchique, en se référant à une organisation de la connaissance en termes de classes et sous-classes. De tels réseaux visent plus particulièrement la modélisation de l'interface monde-lexique à travers la mise en relation de l'univers (profond) des concepts avec l'univers (plus superficiel) des unités lexicales.

*. Cet article est dédié à Anaïs Ferté ; cela fait tellement de mal et tellement de bien, `anafe`, de te croiser au détour du réseau lexical.

2. Les réseaux non ontologiques privilégient quant à eux la modélisation du caractère relationnel des lexiques, en ignorant ou en mettant au second plan la problématique de la classification. Outre les systèmes lexicaux – dont il va être question ici et qui sont manuellement construits –, il s'agit généralement de grands graphes de relations interlexicales (synonymie, inclusion sémantique, etc.) extraits automatiquement des dictionnaires (Ploux & Victorri, 1998) ou d'autres ressources lexicales préexistantes (Spohr, 2012).

2 Le système lexical du Réseau Lexical du Français (RL-fr)

Un système lexical (angl. *lexical system*) d'une langue $\mathcal{L}$ est un modèle du lexique de $\mathcal{L}$ qui possède certaines caractéristiques bien spécifiques de contenu informationnel et de structuration. La notion de système lexical a été théorisée et expérimentée pour la première fois à partir des données lexicographiques de la base lexicale du français DiCo (Polguère, 2000; Mel'čuk & Polguère, 2006) et une description préliminaire du modèle a été publiée dans (Polguère, 2009).

Au moment où a été élaborée la proposition initiale, il n'existait pas d'implémentation de système lexical grandeur nature permettant d'expliciter concrètement le type de modèle postulé et seul un échantillon avait été produit à fin d'illustration. Le RL-fr est la première implantation véritable d'un système lexical et il est désormais possible de s'appuyer sur les résultats obtenus lors de la première étape de construction du RL-fr pour affiner et préciser la notion de système lexical.

Les systèmes lexicaux possèdent quatre caractéristiques principales qui, prises ensemble, les distinguent des autres modèles lexicaux. Elles concernent : les éléments constitutifs du système lexical en tant que graphe – nœuds et arcs (2.1) ; le caractère non atomique de ses nœuds lexicaux (2.2) ; sa structure non ontologique de type graphe petit monde (2.3) ; sa modélisation relativiste de l'information lexicale (2.4).

2.1 Éléments constitutifs

Le système lexical de la langue $\mathcal{L}$ est, formellement, un graphe orienté. Les nœuds de ce graphe correspondent aux différentes entités lexicales de $\mathcal{L}$. Ces entités sont principalement les unités lexicales proprement dites de $\mathcal{L}$, c'est-à-dire :

– les lexèmes de $\mathcal{L}$ – TORCHON **I** [*essuyer avec un torchon*], TORCHON **II** [*Cet article est un torchon.*]. . . ;
– les locutions de $\mathcal{L}$ – ⌜COUP DE TORCHON⌝, ⌜MÉLANGER LES TORCHONS ET LES SERVIETTES⌝. . . [1]

Il peut aussi s'agir, plus marginalement, d'expressions phraséologiques non lexicalisées, notamment de clichés linguistiques comme *Chien méchant* ou *Je vous en prie* [2].

Dans tous les cas, il est fondamental que chaque nœud du graphe soit associé à un sens de $\mathcal{L}$ bien spécifique. Il ne peut s'agir notamment d'un vocable polysémique (grosso modo, une entrée de dictionnaire), qui est dans les faits un regroupement d'unités lexicales (acceptions du vocable). Ainsi, le vocable TORCHON correspond dans le système lexical à un ensemble regroupant deux nœuds lexicaux {TORCHON **I**, TORCHON **II**}. Un système lexical est donc un graphe « désambiguïsé », si on le compare, par exemple, au graphe de synonymie DicoSyn présenté dans (Ploux, 1997). Les nœuds de ce dernier graphe sont en quelque sorte une compaction de copolysèmes et correspondent aux vocables de la langue [3]. Noter que nous désignerons dorénavant les unités lexicales (lexèmes ou locutions) par le terme plus compact de *lexie*.

Les arcs du système lexical correspondent, dans leur immense majorité, à des liens de fonctions lexicales Sens-Texte (Mel'čuk, 1996). Rappelons que le système des fonctions lexicales dites *standard* (Polguère, 2007) permet d'encoder formellement les relations lexicales paradigmatiques de dérivation sémantique – synonymie, antonymie, conversivité, noms d'actants, adjectifs actanciels. . . – et les relations syntagmatiques de cooccurrence collocationnelle – intensificateurs, verbes supports. . . Ainsi, la fonction lexicale **Oper$_1$** correspond aux verbes supports qui prennent le premier actant d'une lexie comme sujet et la lexie elle-même comme complément, tel qu'illustré en (1) ci-dessous avec la lexie HOLD-UP **I**, qui est la lexie de base (non métaphorique) du vocable HOLD-UP.

1. Les noms d'unités lexicales sont écrits en petites majuscules, éventuellement accompagnés de numéros distinctifs en gras. Les noms d'unités lexicales qui sont des locutions sont encadrés par les symboles « ⌜ ⌝ ».
2. Pour une présentation détaillée de la terminologie lexicale utilisée, on peut se reporter à (Mel'čuk *et al.*, 1995).
3. Fort naturellement, cette approche considère comme unité lexicale ce que nous appelons *vocable*. Il est alors possible à S. Ploux et B. Victorri de parler d'*unités lexicales polysémiques* (Ploux & Victorri, 1998). Dans notre terminologie, une telle expression serait antinomique puisque notre unité de description du lexique – l'unité lexicale – est par définition associée à un seul et unique sens.

(1) **Oper$_1$**(HOLD-UP **I**) = *commettre* [ART ∼], *faire* **II.1** [ART ∼], *réaliser* **III** [ART ∼] [4]

La Figure 1 est la traduction sous forme de micrographe lexical de l'application de fonction lexicale (1).

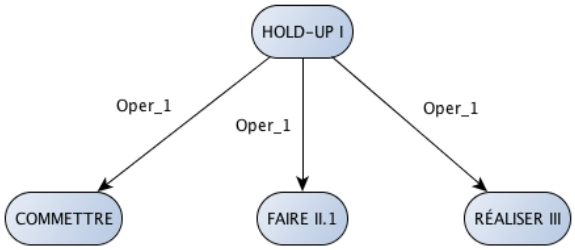


FIGURE 1 – Micrographe représentant **Oper$_1$**(HOLD-UP **I**)

Attention cependant ! La Figure 1 correspond à la modélisation de (1) selon les principes de structuration des systèmes lexicaux. Sur le plan de l'informatisation de ces derniers, cependant, rien ne nous oblige à implanter directement de tels graphes. Ainsi, tel que décrit dans (Gader *et al.*, 2012), le système lexical du RL-fr est implanté sous la forme d'une base SQL extrêmement modulaire permettant de gérer efficacement le processus complexe d'édition lexicographique. Nous fonctionnons alors par associations entre entités élémentaires de la base. Dans le cas des liens de fonctions lexicales, une application de fonction telle que celle de (1) est modélisée comme : une association entre l'ID d'une lexie donnée (HOLD-UP **I**) avec l'ID d'une fonction lexicale donnée (**Oper$_1$**), association dont l'ID pointe elle-même vers les ID des trois lexies cibles de l'application de la fonction lexicale (COMMETTRE, FAIRE **II.1** et RÉALISER **III**). Trois tables SQL distinctes sont principalement mises à l'œuvre pour encoder ces connexions : (i) `ln_senses`, qui est la table des entités lexicales nœuds du graphe, (ii) `ln_lf`, qui est la table des fonctions lexicales du RL-fr et (iii) `ln_senses_lf`, qui est la table des associations entre les ID des entités de `ln_senses` et les ID des entités de `ln_lf` [5].

Un tel encodage se visualise en terme de graphe par la Figure 2 ci-dessous, ce qui correspond à une structure radicalement distincte de celle de la Figure 1.

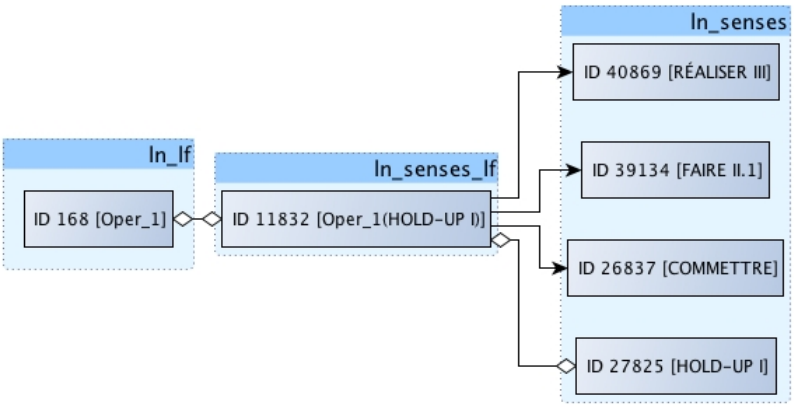


FIGURE 2 – Implantation dans le RL-fr du micrographe de la Figure 1

4. Les numéros distinctifs utilisés pour identifier les sens lexicaux dans cet article correspondent à ceux en vigueur dans le RL-fr au moment où nous écrivons. Ainsi, l'absence de numérotation après *commettre* découle du fait que la polysémie du vocable COMMETTRE n'est pas encore dégagée et qu'à l'heure actuelle seule existe dans le RL-fr l'acception correspondant à l'emploi en tant que verbe support de type **Oper$_1$** (*commettre un hold-up*, *un péché*, *une erreur*...).

5. En d'autres termes, il s'agit de la table qui modélise les applications de fonctions lexicales à des lexies spécifiques. Dans les noms de tables SQL, `ln` est mis pour *lexical network* et `lf` pour *lexical function*.

Cet article vise avant tout la présentation des systèmes lexicaux en tant que « théorisation » de la structure des lexiques, à partir des acquis du travail de construction du RL-fr. Dans la pratique cependant, il convient de choisir le type de structure informatique le plus approprié à l'usage que l'on compte faire des données du réseau lexical. Comme mentionné ci-dessus, nous avons utilisé une structure de type SQL pour répondre aux besoins particuliers de l'édition lexicographique : opérations de création/destruction de nœuds, tissage/détissage de liens, ajout d'informations riches associées à chaque nœud, etc., qui sont effectuées par une équipe complète de lexicographes travaillant simultanément sur les mêmes données. Des structures XML, RDF, etc., peuvent être préférées pour d'autres usages [6].

Les liens paradigmatiques et syntagmatiques de fonctions lexicales Sens-Texte forment l'ossature principale des systèmes lexicaux, mais ils ne sont pas les seuls à tisser les relations lexicales de la langue. On trouve ainsi notamment au sein des systèmes lexicaux les trois types de liens suivants entre nœuds du graphe :

1. liens de copolysémie – par exemple, un lien `métaphore : comme si` connecte la lexie HOLD-UP **II** [*Ce vote est un hold-up démocratique.*] à la lexie de base HOLD-UP **I** ;
2. liens asémantiques entre les locutions et les lexies dont elles sont formellement constituées – par exemple, la locution ⌜DONNER CORPS⌝ [*donner corps à une idée*] pointe vers les lexies DONNER **III.2** et CORPS **I.1a** ;
3. liens d'inclusion sémantique définitionnelle – par exemple la lexie HOLD-UP **I** contient dans sa définition le sens de la lexie VOLER **2 I**.

Pour l'instant, seuls les deux premiers types de liens additionnels énumérés ci-dessus – copolysémie et inclusion formelle dans les locutions – sont implantés dans le RL-fr. Le travail d'encodage des liens d'inclusion sémantique définitionnelle vient à peine de débuter.

Comme on le voit à l'examen des différents profils de nœuds et arcs présents dans un système lexical, ce modèle a notamment la particularité de correspondre à une structure de graphe relativement hétérogène. Nous considérons que cette hétérogénéité, si elle peut être vue comme un handicap sur le plan formel, n'est en fait que le reflet d'une propriété intrinsèque de l'organisation du lexique des langues.

Pour conclure ce premier niveau de caractérisation des systèmes lexicaux, nous résumons dans le Tableau 1 ci-dessous les statistiques actuelles du RL-fr en termes de nombre de nœuds et d'arcs constitutifs du graphe.

Nombre de nœuds	22 824
Nombre d'arcs de liens de fonctions lexicales	40 361
Nombre d'arcs de copolysémie	2 898
Nombre d'arcs « locution→lexies formellement incluses »	5 028
Degré moyen des nœuds (taux de connectivité)	2,12

Tableau 1 – Statistiques actuelles du RL-fr en termes de nœuds et d'arcs

Il convient de noter que le tissage des liens de copolysémie a débuté récemment, ce qui explique le fait que très peu d'entre eux soient pour l'instant présents dans le RL-fr. On remarque aussi dans les statistiques du Tableau 1 que la taille du RL-fr commence à être significative, en termes d'entités lexicales décrites (principalement, des lexies), mais que le taux de connectivité est encore bas : certaines lexies bien décrites sont fortement connectées, alors que d'autres sont presque isolées dans le graphe. Le taux de connectivité croît cependant de façon constante.

2.2 Caractère non atomique des nœuds lexicaux

Dans un système lexical, les nœuds lexicaux du graphe ne sont pas atomiques, en ce sens qu'ils possèdent eux-mêmes une structure interne complexe. « Ouvrir » un nœud lexical pour regarder ce qu'il y a à l'intérieur revient en quelque sorte à afficher l'article lexicographique correspondant. Chaque nœud représentant une lexie est ainsi associé à un ensemble structuré de propriétés, qui sont principalement :

– les caractéristiques grammaticales de la lexie – parties du discours, genre (pour les noms), tables de flexion, etc. ;
– son sens – étiquette sémantique (Polguère, 2011), forme propositionnelle (c.-à-d., structure actancielle) et définition proprement dite ;

6. Ainsi, les visualisations présentées dans la section 3.2 ci-dessous sont générées à partir de graphes construits en Python à l'aide de la bibliothèque `igraph`. (Merci à Yann Desalle pour ces informations.)

– son régime syntaxique ;
– ses connexions paradigmatiques et syntagmatiques – directement implantées par le tissage des liens du graphe lexical ;
– des illustrations de ses emplois – exemples lexicographiques.

Ces ensembles de propriétés correspondent aux différentes zones d'un article de lexicographique d'un *Dictionnaire Explicatif et Combinatoire* (Mel'čuk et al., 1984 1988 1992 1999) ou de toute base lexicale conçue selon les principes de la Lexicologie Explicative et Combinatoire. La Figure 3 illustre ce fait en présentant un extrait de la *vue-article* de HOLD-UP **I** dans son état courant, telle que générée par l'éditeur lexicographique Dicet (Gader *et al.*, 2012) [7].

FIGURE 3 – Extrait de la vue-article de HOLD-UP **I** dans l'éditeur Dicet

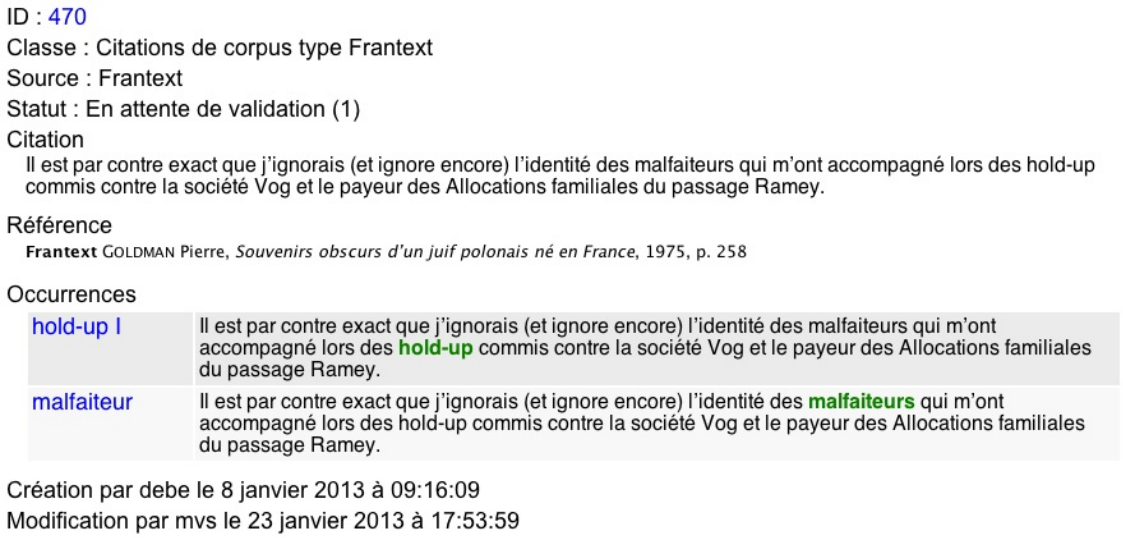
La vue-article d'une lexie est la rétroaction de forme textuelle que l'éditeur Dicet génère pour le lexicographe à partir des informations linguistiques associées à la lexie en question. Il ne s'agit donc aucunement d'un texte éditable par le lexicographe. Ce dernier n'écrit pas d'article lexicographique et son activité principale consiste à « tisser sa toile » (Polguère, 2012b). Cette approche lexicographique – suivant laquelle le texte dictionnairique n'est pas écrit, mais est virtuellement encapsulé dans le modèle lexical (Atkins, 1996; Polguère, 2012a) – nous semble celle qui se prête le mieux à la construction des systèmes lexicaux du fait de la nature fondamentalement non textuelle de ces derniers.

Mentionnons, pour conclure cette section, que les propriétés « internes » des nœuds lexicaux sont, comme leurs propriétés relationnelles, tout à fait modélisables sous forme de graphe. Dans le cas du RL-fr, chaque propriété associée à un nœud lexical est implantée comme un pointeur vers une entité autonome de la base. Indiquer qu'une lexie est un nom commun, est du genre masculin ou est familière revient à tisser un lien entre la lexie en question et des entités métalinguistiques stockées dans des modèles intégrés des caractéristiques grammaticales, des étiquettes sémantiques, des fonctions lexicales, etc. Le RL-fr pousse en fait cette approche à son extrême, puisque même les exemples lexicographiques y sont des entités autonomes vers lesquelles pointent les lexies individuelles [8]. Plusieurs lexies peuvent ainsi pointer vers le même exemple,

7. Le régime syntaxique (structure de complémentation) n'apparaît pas pour l'instant dans la vue-article.

8. L'ensemble des citations utilisées dans le RL-fr constitue en soi un corpus linguistique enchâssé dans le modèle. Au moment où nous écrivons, le corpus des citations du RL-fr totalise plus d'un million de mots-occurrences ; il est notamment accessible par les lexicographes au moyen de concordances effectuées avec le logiciel TXM (Heiden *et al.*, 2010). Le corpus enchâssé des citations ne doit bien entendu pas être confondu avec les différents corpus de référence dont les lexicographes font usage et dont les citations du RL-fr sont justement extraites.

chaque pointeur visant une séquence linguistique spécifique dans l'exemple ; cela permet entre autres d'ouvrir la « fiche lexicographique » d'un exemple donné et d'accéder à partir d'elle à toutes les lexies qui y font appel. C'est ce qu'illustre la Figure 4 ci-dessous, qui est la visualisation sous Dicet de l'information associée au second exemple lexicographique apparaissant dans la Figure 3 (article de HOLD-UP **I**).


ID : 470
Classe : Citations de corpus type Frantext
Source : Frantext
Statut : En attente de validation (1)
Citation
Il est par contre exact que j'ignorais (et ignore encore) l'identité des malfaiteurs qui m'ont accompagné lors des hold-up commis contre la société Vog et le payeur des Allocations familiales du passage Ramey.
Référence
Frantext GOLDMAN Pierre, *Souvenirs obscurs d'un juif polonais né en France*, 1975, p. 258
Occurrences

hold-up I	Il est par contre exact que j'ignorais (et ignore encore) l'identité des malfaiteurs qui m'ont accompagné lors des **hold-up** commis contre la société Vog et le payeur des Allocations familiales du passage Ramey.
malfaiteur	Il est par contre exact que j'ignorais (et ignore encore) l'identité des **malfaiteurs** qui m'ont accompagné lors des hold-up commis contre la société Vog et le payeur des Allocations familiales du passage Ramey.

Création par debe le 8 janvier 2013 à 09:16:09
Modification par mvs le 23 janvier 2013 à 17:53:59

FIGURE 4 – Fiche d'un exemple lexicographique dans l'éditeur Dicet

2.3 Structure non ontologique de type petit monde

Comme mentionné dans l'Introduction, les systèmes lexicaux sont des modèles non ontologiques. Cela découle directement du fait que l'ossature de ces modèles est tissée à partir du système des fonctions lexicales : il s'agit d'un système de relations sémantico-syntaxiques qui se distingue radicalement de l'organisation hiérarchique des concepts, en classes et sous-classes, postulées par les modèles ontologiques. Pour dire les choses de façon caricaturale : la visualisation d'un lexique ontologique évoque une sorte de tour Eiffel aplatie, alors que celle d'un système lexical présente plutôt une analogie avec un plat de spaghetti.

Le graphe d'un système lexical peut ainsi sembler être un « fouillis » de relations paradigmatiques et syntagmatiques où n'émerge aucune structure classifiante. Pourtant, un tel graphe possède au moins quatre propriétés formelles remarquables [9] :

P1 La densité en arcs est faible : les graphes lexicaux, et les systèmes lexicaux en particulier, possèdent peu d'arcs, en regard de leur nombre total de sommets.

P2 La moyenne des plus courts chemins est petite : en général, il existe au moins un chemin relativement court entre n'importe quelle paire de sommets.

P3 Il existe des *clusters* : les graphes lexicaux sont localement denses en arcs, alors que leur densité globale est faible (propriété **P1**).

P4 La distribution du degré d'incidence des sommets suit une loi de puissance : la probabilité qu'un sommet donné ait k voisins décroit comme une loi de puissance de k (alors que c'est une loi de Poisson dans les graphes aléatoires).

Les graphes possédant les propriétés **P1**, **P2** et **P3** sont appelés *graphes petits mondes* (angl. *small-world networks/graphs*), et ont été mathématiquement définis dans (Watts & Strogatz, 1998).

9. La formulation qui suit nous a été suggérée par Bruno Gaume, que nous remercions pour nous avoir aidé à mettre plus de rigueur mathématique dans nos propos.

Les propriétés des petits mondes lexicaux ont été systématiquement étudiées par Bruno Gaume (Gaume, 2004; Gaume *et al.*, 2006; Gaume, 2008), qui a démontré qu'il était possible d'exploiter leurs particularités formelles pour y identifier des espaces sémantiques structurés et, de façon plus générale, pour mettre de l'ordre dans le désordre lexical apparent.

Le RL-fr, même s'il s'agit d'un réseau lexical encore peu développé, est déjà suffisamment riche pour manifester les caractéristiques des graphes petits mondes (Gader *et al.*, 2014). En théorie, tout système lexical, tel que défini ici, devra posséder ce type de structuration.

2.4 Modèle relativiste

Les systèmes lexicaux incorporent une mesure de flou. Cette propriété des systèmes lexicaux a été anticipée dès la proposition initiale de (Polguère, 2009) pour deux raisons :

1. La connaissance lexicale est elle-même gradable, du fait notamment des changements diachroniques et de l'incertitude linguistique potentielle des locuteurs, y compris des lexicographes.
2. Les systèmes lexicaux doivent être des modèles supportant l'inférence et la construction de la connaissance lexicale par approximation – cf. plus bas, section 3.3.

Le RL-fr implémente cette nature gradable de l'information lexicale. Chaque information linguistique – existence même d'une lexie, ses caractéristiques grammaticales, etc. – est associée à un niveau de confiance. Ce dernier est par défaut maximal lorsque l'information est entrée manuellement par les lexicographes. Ceux-ci peuvent cependant « nuancer » l'information en lui associant un niveau de confiance moindre. Par convention, la confiance minimale qui peut être manuellement entrée est de 60%. Toute information qui sera automatiquement injectée dans la base se verra quant à elle attribuer un niveau de confiance de 50%, et c'est cette confiance spécifique relativement basse, associée aux données résultant d'un enrichissement automatique de la base, qui permet leur identification.

Nous avons ainsi généré de façon automatique un Réseau Lexical de l'Anglais, RL-en, par compilation de l'information lexicographique contenue dans le WordNet de Princeton (Fellbaum, 1998) [10]. Toutes les informations du système lexical résultant sont associées à un niveau de confiance de 50% et toute entrée de données qui sera effectuée manuellement sur le RL-en au moyen de l'éditeur Dicet générera par contraste une information validée (niveau de confiance de 100%).

Bien entendu, nous ne faisons pour l'instant qu'un usage un peu caricatural des mesures de confiance associées aux données du RL-fr et du RL-en. Ces ressources sont cependant équipées de tous les dispositifs qui permettront de tirer parti de cette propriété des systèmes lexicaux, notamment pour ce qui est de la semi-automatisation de la croissance de la ressource, sur laquelle nous revenons dans la section 3.3.

3 Justifications de l'approche

Dans cette section, nous tenterons d'apporter plusieurs pistes de justification théorique et empirique pour l'adoption des systèmes lexicaux en tant que modèles formels des lexiques. Nous abordons successivement les trois points suivants : possible intérêt des systèmes lexicaux dans une perspective psycholinguistique (3.1) ; application de l'analyse proxémique (3.2) ; semi-automatisation du processus de croissance du RL-fr et, plus généralement, des systèmes lexicaux (3.3).

3.1 Les systèmes lexicaux comme modèles du lexique mental

On entend par *lexique mental* (angl. *mental lexicon*) la connaissance lexicale telle qu'elle est stockée et structurée dans le cerveau d'un individu (Aitchison, 2003; Zock, 2005; Wierzbicka, 2009). Il existe deux raisons principales pour lesquelles il serait intéressant que les systèmes lexicaux soient une représentation plausible de la structure des lexiques mentaux. Tout d'abord, cela ferait des systèmes lexicaux un support pertinent pour mener des études aussi bien lexicologiques que psycholinguistiques, les modèles construits étant en harmonie avec le « lexique véritable ». Ensuite, on peut faire l'hypothèse que de tels modèles permettraient de simuler informatiquement les tâches d'accès et de manipulation de l'information lexicale avec la même efficacité que le fait le cerveau humain. La recherche d'une adéquation entre modèle formel construit et lexique mental postulé peut donc trouver sa justification aussi bien sur un plan théorique que pratique.

10. La méthode de génération automatique du RL-en est décrite dans (Gader *et al.*, 2014).

On notera que l'adéquation au lexique mental était justement la visée première du projet WordNet de Princeton et que (Miller *et al.*, 1990) se situaient explicitement dans un contexte de *psycholexicologie* (angl. *psycholexicology*). Comme le relève (Zock, 2005, p. 113) cependant, « [...] les psycholinguistes travaillant sur l'accès lexical ne mentionnent pour ainsi dire jamais WordNet. La communauté de WordNet, malgré sa taille, semble complètement ignorer les travaux sur l'accès lexical ». Au-delà de la seule problématique de l'accès lexical, c'est toute la perspective psycholinguistique qui est quasi absente des travaux menés sur et à partir de WordNet [11]. Ces travaux exploitent dans leur grande majorité la très haute qualité lexicographique de WordNet (plutôt que sa validité psycholinguistique postulée) dans un contexte de Traitement Automatique de la Langue.

Tout cela ne veut bien entendu pas dire que WordNet n'est pas psycholinguistiquement pertinent. Simplement, sa pertinence sur ce plan est loin d'être démontrée et il est, par conséquent, tout à fait légitime d'explorer d'autres avenues. Nous postulons quant à nous, sans pouvoir le démontrer pour l'instant et sans même prétendre nous appuyer sur des acquis de la psychologie moderne [12], que l'approche des systèmes lexicaux est compatible avec des exploitations psycholinguistiques. Nous voyons les systèmes lexicaux comme une approximation de la structure des lexiques mentaux, si l'on entend par ce terme spécifiquement la connaissance linguistique lexicale. Nous ne sommes bien entendu aucunement en mesure d'émettre quelque hypothèse que ce soit sur la façon dont les concepts eux-mêmes et la connaissance extra-linguistique sont mentalement organisés.

À l'appui du recours aux systèmes lexicaux pour l'étude du lexique mental, nous nous contenterons de mentionner la compatibilité formelle de ces derniers avec de nombreuses propositions faites pour la modélisation de l'acquisition du vocabulaire en tant qu'enrichissement de réseaux lexicaux – voir, par exemple, (Wolter, 2006) et (Gaume *et al.*, 2008). Les domaines d'application qui méritent d'être explorés prioritairement pour valider l'hypothèse d'une pertinence psycholinguistique des systèmes lexicaux nous semblent être : l'analyse des erreurs lexicales en parole spontanée et le diagnostic de troubles mentaux ou physiologiques se manifestant sur le plan de l'accès lexical.

3.2 Identification d'espaces sémantiques dans les systèmes lexicaux

Nous entendons ici par *espace sémantique* d'un réseau lexical représentant le lexique d'une langue $\mathcal{L}$ un ensemble plutôt petit d'entités lexicales de $\mathcal{L}$ présentant une forme de cohérence sémantique. La notion d'espace sémantique est proche de celle de champ sémantique, mais elle s'en distingue par le fait que tous les éléments d'un tel espace n'appartiennent pas nécessairement à un champ sémantique donné [13]. En effet, le sens d'une lexie est directement connecté à sa combinatoire et les lexies sémantiquement proches tendent à partager de façon significative des éléments de combinatoire. Cela vaut bien entendu pour ce que l'on appelle la *combinatoire libre*, qui découle directement de la définition de la lexie, mais aussi pour la combinatoire lexicale restreinte. Ainsi, même si l'on doit considérer qu'un syntagme comme *commettre un hold-up* – analysé plus haut, section 2.1 – est une collocation de type **Oper$_1$** contrôlée par la lexie HOLD-UP **I**, il est logique de postuler que nombre de lexies nominales sémantiquement proches de HOLD-UP **I** tendront à contrôler le même **Oper$_1$** : *commettre un crime*, *un massacre*, *un meurtre*, *un vol*, etc. Les lexies d'un même champ sémantique ont donc tendance non seulement à former des « communautés » au sein des graphes lexicaux petits mondes, mais elles ont aussi tendance à exercer une force d'attraction sur des lexies collocationnelles très vagues ou quasi vides et à les aspirer dans leur espace sémantique. Un espace sémantique est donc potentiellement une expansion d'un champ sémantique résultant des affinités lexicales syntagmatiques.

Le fait que les systèmes lexicaux traitent de façon uniforme les liens paradigmatiques et les liens syntagmatiques au sein d'une même structure va permettre de tirer parti de la combinatoire des lexies pour identifier de façon encore plus efficace les espaces sémantiques au sein des lexiques, notamment par le recours à la proxémie. Cette dernière est une mesure de similarité définie sur les graphes petits mondes par B. Gaume (Gaume, 2004) et qui a tout particulièrement été appliquée aux graphes de synonymie (Gaume *et al.*, 2006). Elle permet d'identifier des espaces sémantiques de façon entièrement « non linguistique », c'est-à-dire en s'appuyant exclusivement sur la structure du graphe « nu », exprimée en termes de sommets et d'arcs connectant ces sommets.

L'analyse proxémique par la méthode de parcours Prox (Gaume, 2008) est présentement appliquée sur le RL-fr, sans ajustement particulier, pour en extraire une clusterisation du système lexical à partir de lexies données. Une visualisation par Tmuse (Chudy *et al.*, 2013) permet de représenter graphiquement les résultats obtenus, comme illustré dans la Figure 5 ci-dessous à partir de la lexie FOOTBALL **1** du RL-fr.

11. Ainsi, aucun article présenté au dernier congrès *Global WordNet* (Orav *et al.*, 2014) ne concerne une recherche relevant de la psycholinguistique.
12. Voir néanmoins (Morais *et al.*, 2013).
13. Pour une définition de la notion de champ sémantique, en contraste notamment avec celle de champ lexical, voir (Polguère, 2013).

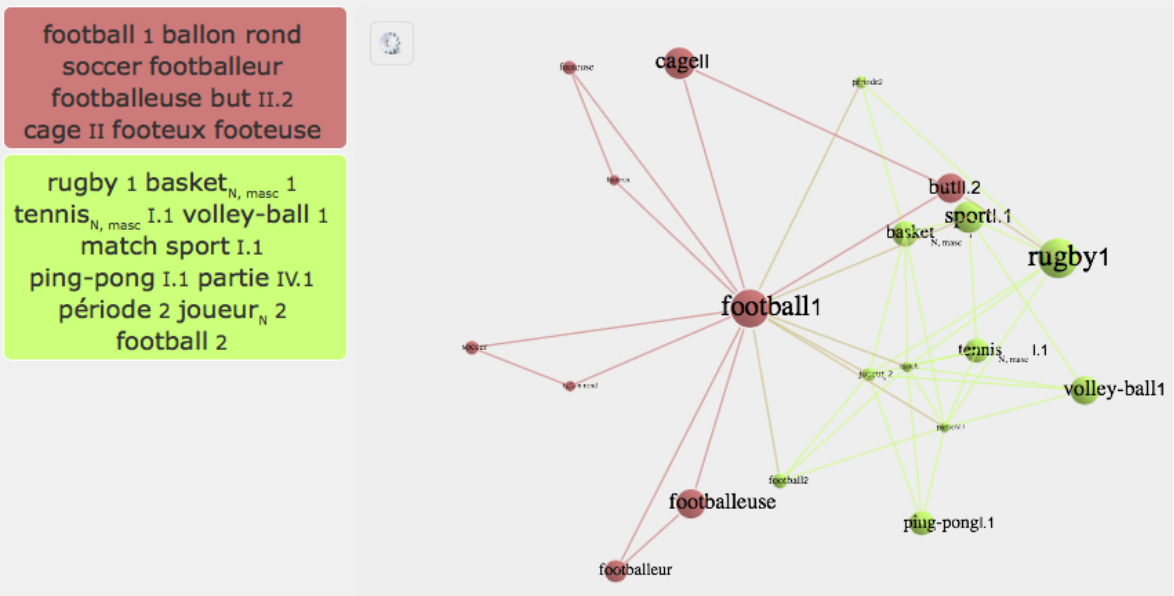


FIGURE 5 – Clusterisation par proxémie et visualisation de FOOTBALL **1**

On voit que les deux clusters qui ont été ici identifiés par Prox, sans recours à la sémantique même des liens de fonction lexicale, correspondent à une segmentation de l'espace sémantique de la lexie selon une base assez logique, qui est grosso modo : vocabulaire du football *vs* vocabulaire connecté à celui du football. On notera que FOOTBALL **2**, qui apparaît dans le second cluster, aurait en réalité tout à fait sa place dans le premier, puisque cette lexie appartient très directement au vocabulaire du football : FOOTBALL **2** est l'acception métonymique du vocable, que l'on trouve par exemple dans *Tu viens faire un petit football dans la ruelle ?* Cette imperfection dans le résultat obtenu doit possiblement être interprétée comme un indice du fait que la connexion de FOOTBALL **2** au reste du graphe est encore incomplète. Nous n'en sommes qu'au tout début de l'application des analyses proxémiques aux systèmes lexicaux. Il s'agit d'un travail qui va se poursuivre sur le long terme, dans une visée aussi bien applicative que descriptive, comme nous allons maintenant l'expliquer.

3.3 Vers une semi-automatisation de la croissance du RL-fr

Il existe quatre principales méthodes de construction des ressources lexicales destinées au Traitement Automatique de la Langue ou à tout autre domaine d'application langagier :

1. la méthode lexicographique, c'est-à-dire la construction manuelle des ressources lexicales telle qu'elle est pratiquée pour le RL-fr ou telle qu'elle a été pratiquée dès le lancement du projet WordNet (Miller *et al.*, 1990) ;
2. la méthode coopérative, qui repose sur une approche type *crowdsourcing* telle qu'appliquée pour Wiktionary [14] ou, selon une stratégie ludique, avec JeuxDeMots (Lafourcade & Joubert, 2013) [15] ;
3. la méthode automatique à base de ressources lexicales, qui consiste en l'extraction et la structuration informatique de données linguistiques à partir de ressources lexicales préexistantes, comme dans le cas de WOLF (Sagot & Fišer, 2008) ou de BabelNet (Navigli & Ponzetto, 2012) ;
4. la méthode automatique à base de corpus textuels, qui s'appuie exclusivement sur des données linguistiques brutes, et non sur des modélisation métalinguistiques antérieurement construites (Hearst, 1992).

On notera que la troisième méthode présuppose l'activité de linguistes et lexicographes. Sans dictionnaires et autres WordNet, il n'y aurait, par définition, pas de matériau permettant d'alimenter les projets de construction automatique de ressources dérivées de ces ressources mères. Ces ressources mères ne peuvent d'ailleurs jamais être considérées comme

14. http://www.wiktionary.org
15. http://www.jeuxdemots.org/jdm-accueil.php

ayant été produites une fois pour toutes : la langue évolue et la modélisation lexicographique est dans les faits un travail perpétuellement en cours.

La seconde approche est exclue si l'on veut disposer de ressources proposant une description fine de l'information lexicale et d'une qualité qui permette, par exemple, de l'utiliser dans un contexte pédagogique. De même qu'un patient a des raisons de s'inquiéter lorsqu'il voit son médecin chercher sur Wikipedia de l'information sur une maladie ou un symptôme le concernant, de même un parent d'élève du primaire serait en droit de s'inquiéter de voir l'enseignant de son enfant utiliser Wiktionary, plutôt qu'un dictionnaire élaboré par des lexicographes, pour appuyer son enseignement du vocabulaire.

Il faut donc bien que « quelqu'un s'y colle », et c'est justement la tâche (et le plus grand plaisir) des lexicographes que de produire les ressources lexicales dont, ultimement, tout le monde a besoin.

En dépit de cela, il est plus que légitime d'envisager une part d'automatisation du processus lexicographique. Cela peut aller de l'utilisation d'outils proposant des ébauches de description (Kilgarriff & Rychlý, 2010) à l'enrichissement semi-automatique par inférences fondées sur la structure même de l'information lexicale (Sajous *et al.*, 2011). Il nous semble que la structure des systèmes lexicaux se prête tout particulièrement à ce type de semi-automatisation du travail lexicographique. Les premières observations qui ont été faites sur le traitement automatique du graphe lexical du RL-fr par la proxémie (section 3.2, ci-dessus) sont de ce point de vue encourageantes. La clusterisation permet dans de nombreux cas de mettre en évidence des trous dans l'espace sémantique des lexies, trous résultant de la jeunesse du RL-fr. Il peut s'agir de lexies manquantes dans le modèle ou, beaucoup plus fréquemment, d'une connexion insuffisante au reste du graphe des lexies déjà présentes. Il devient alors intéressant de tenter d'exploiter les algorithmes de parcours de graphes petits mondes pour repérer dans le RL-fr des cliques ou des « presque-cliques » [16] significatives et identifier des micropatrons récurrents de connexions lexicales. Ces patrons pourront ensuite être exploités afin de générer automatiquement des éléments de graphe plausibles [17]. Ces derniers, qui devront être validés manuellement, permettront d'obtenir une sorte de « RL-fr turbo » constitué d'un noyau entièrement validé, pour l'exploitation humaine, et d'une extension à plus faible taux de confiance, pour compléter le RL-fr noyau dans un contexte de Traitement Automatique de la Langue notamment.

4 Conclusion

On ne peut bien entendu pas considérer la pertinence des systèmes lexicaux comme étant un fait acquis, tant que l'efficacité de ces modèles n'aura pas été démontrée à travers leur mise en pratique. Pour cela, il va falloir expérimenter, notamment dans les domaines de la recherche d'information, de la didactique des langues ou de l'étude psycholinguistique. Cependant, il est d'ores et déjà justifié de dire qu'une première étape essentielle dans la validation de l'approche des systèmes lexicaux a été franchie. Il a en effet été démontré, à travers le travail de construction du RL-fr, que le mode de structuration offert par les systèmes lexicaux permet de mener avec efficacité un programme lexicographique véritable directement conçu comme le tissage d'un graphe lexical [18].

Remerciements

Le projet RELIEF, auquel participe la société MVS de Sainte-Marguerite, est financé par l'Agence de Mobilisation Économique de la Région Lorraine et le FEDER Lorrain. Les résultats présentés ici ont été obtenus grâce au travail considérable effectué par les lexicographes du RL-fr (`http://www.atilf.fr/spip.php?article908`) et par Nabil Gader, développeur de l'éditeur lexicographique Dicet ; nous les remercions tous chaleureusement. Le travail en cours sur la visualisation du graphe du RL-fr et la clusterisation par proxémie est effectué par Yann Desalle, en collaboration avec Bruno Gaume et la Proxteam de Toulouse (`http://www.irit.fr/Proxteam/`). Merci à Bruno Gaume, Sandrine Ollinger et à trois relecteurs anonymes de TALN 2014 pour leurs commentaires sur une première version de cet article.

16. Nous entendons par là des sous-graphes qui ne sont pas strictement parlant des cliques (sous-graphes complets), mais dont on peut considérer que le nombre maximal d'arcs est atteint en regard de contraintes spécifiques associées aux données que le graphe modélise. Par exemple, tel sous-graphe d'un système lexical n'est pas une clique, car les deux sommets s_1 et s_2 ne sont pas liés ; **mais** on peut considérer que c'est une presque-clique, car les deux unités lexicales représentées par ces sommets ne devraient de toute façon pas être liées par des liens paradigmatiques ou syntagmatiques, pour des raisons linguistiques.

17. Cf. la recherche doctorale de S. Ollinger, en cours à l'ATILF : *Le raisonnement analogique en lexicographie et son informatisation : application au Réseau Lexical du Français*.

18. Le graphe lexical du RL-fr doit être rendu disponible par téléchargement libre d'ici la fin du projet RELIEF (automne 2014), ce qui doit représenter la fin de première étape dans construction de la ressource. D'ici là, il est bien entendu possible à tout chercheur désirant conduire des expérimentations avec les données du RL-fr de nous contacter directement.

Références

AITCHISON J. (2003). *Words in the Mind : An Introduction to the Mental Lexicon*. Oxford, G.-B. : Blackwell.

ATKINS B. T. S. (1996). Bilingual Dictionaries : Past, Present and Future. In M. GELLERSTAM, J. JÄRBORG, S.-G. MALMGREN, K. NORÉN, L. ROGSTRÖM & C. R. PAPMEHL, Eds., *Euralex'96 Proceedings*, p. 515–590, Gothenburg : Gothenburg University, Department of Swedish.

CHUDY Y., DESALLE Y., GAILLARD B., GAUME B., MAGISTRY P. & NAVARRO E. (2013). Tmuse : Lexical Network Exploration. In *The Companion Volume of the Proceedings of IJCNLP 2013 : System Demonstrations*, p. 41–44, Nagoya, Japan : Asian Federation of NLP.

C. FELLBAUM, Ed. (1998). *WordNet : An Electronic Lexical Database*. Cambridge, MA : The MIT Press.

GADER N., LUX-POGODALLA V. & POLGUÈRE A. (2012). Hand-Crafting a Lexical Network With a Knowledge-Based Graph Editor. In *Proceedings of the Third Workshop on Cognitive Aspects of the Lexicon (CogALex III)*, p. 109–125, Mumbai : The COLING 2012 Organizing Committee.

GADER N., OLLINGER S. & POLGUÈRE A. (2014). One lexicon, two structures : So what gives ? In H. ORAV, C. FELLBAUM & P. VOSSEN, Eds., *Proceedings of the Seventh Global Wordnet Conference (GWC2014)*, p. 163–171, Tartu : Global WordNet Association.

GAUME B. (2004). Balades aléatoires dans les Petits Mondes Lexicaux. *I3 Information Interaction Intelligence*, **4**(2), 39–96. CEPADUES édition.

GAUME B. (2008). Mapping the Forms of Meaning in Small Worlds. *Journal of Intelligent Systems*, **23**, 848–862.

GAUME B., DUVIGNAU K., PRÉVOT L. & DESALLE Y. (2008). Toward a cognitive organization for electronic dictionaries, the case for semantic proxemy. In *Proceedings of the workshop on Cognitive Aspects of the Lexicon (COGALEX 2008)*, p. 86–93, Manchester.

GAUME B., VENANT F. & VICTORRI B. (2006). Hierarchy in Lexical Organization of Natural Languages. In D. PUMAIN, Ed., *Hierarchy in Natural and Social Sciences*, Methodos series 3, p. 121–142. Dordrecht : Springer.

HEARST M. A. (1992). Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING 1992)*, p. 539–545, Nantes.

HEIDEN S., MAGUÉ J.-P. & PINCEMIN B. (2010). TXM : Une plateforme logicielle open-source pour la textométrie – conception et développement. In S. BOLASCO, I. CHIARI & L. GIULIANO, Eds., *Statistical Analysis of Textual Data – Proceedings of 10th International Conference Journées d'Analyse statistique des Données Textuelles*, volume 2-3, p. 1021–1032, Rome : Edizioni Universitarie di Lettere Economia Diritto.

KILGARRIFF A. & RYCHLÝ P. (2010). Semi-automatic Dictionary Drafting. In G.-M. DE SCHRYVER, Ed., *A Way with Words : Recent Advances in Lexical Theory and Analysis. A Festschrift for Patrick Hanks*, p. 299–312. Kampala : Menha Publishers.

LAFOURCADE M. & JOUBERT A. (2013). Bénéfices et limites de l'acquisition lexicale dans l'expérience jeuxdemots. In N. GALA & M. ZOCK, Eds., *Ressources Lexicales : Contenu, construction, utilisation, évaluation*, LingvisticæInvestigationes, Supplementa 30, p. 187–216. Amsterdam/Philadelphia : John Benjamins.

LUX-POGODALLA V. & POLGUÈRE A. (2011). Construction of a French Lexical Network : Methodological Issues. In *Proceedings of the First International Workshop on Lexical Resources, WoLeR 2011. An ESSLLI 2011 Workshop*, p. 54–61, Ljubljana.

MEL'ČUK I. (1996). Lexical Functions : A Tool for the Description of Lexical Relations in the Lexicon. In L. WANNER, Ed., *Lexical Functions in Lexicography and Natural Language Processing*, volume 31 of *Language Companion Series*, p. 37–102. Amsterdam/Philadelphia : John Benjamins.

MEL'ČUK I. (2006). Explanatory Combinatorial Dictionary. In G. SICA, Ed., *Open Problems in Linguistics and Lexicography*, p. 225–355. Monza : Polimetrica.

MEL'ČUK I., CLAS A. & POLGUÈRE A. (1995). *Introduction à la lexicologie explicative et combinatoire*. Paris/Louvain-la-Neuve : Duculot.

MEL'ČUK I. & POLGUÈRE A. (2006). Dérivations sémantiques et collocations dans le DiCo/LAF. *Langue française*, **150**, 66–83.

MEL'ČUK ET AL. I. (1984, 1988, 1992, 1999). *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques. Volumes I–IV*. Montréal : Les Presses de l'Université de Montréal.

MILLER G. A., BECKWITH R., FELLBAUM C., GROSS D. & MILLER K. J. (1990). Introduction to WordNet : An On-line Lexical Database. *International Journal of Lexicography*, **3**(4), 235–244.

MORAIS A. S., OLSSON H. & SCHOOLER L. J. (2013). Mapping the Structure of Semantic Memory. *Cognitive Science*, **37**, 125–145.

NAVIGLI R. & PONZETTO S. P. (2012). BabelNet : The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. *Artificial Intelligence*, **193**, 217–250.

H. ORAV, C. FELLBAUM & P. VOSSEN, Eds. (2014). *Proceedings of the Seventh Global Wordnet Conference (GWC2014)*, Tartu (Estonia). Global WordNet Association.

PLOUX S. (1997). Modélisation et traitement informatique de la synonymie. *Lingvisticæ Investigationes*, **21**(1), 1–27.

PLOUX S. & VICTORRI B. (1998). Construction d'espaces sémantiques à l'aide de dictionnaires de synonymes. *Traitement Automatique des Langues (T.A.L.)*, **39**(1), 161–182.

POLGUÈRE A. (2000). Towards a theoretically-motivated general public dictionary of semantic derivations and collocations for French. In *Proceedings of EURALEX'2000*, p. 517–527, Stuttgart.

POLGUÈRE A. (2007). Lexical function standardness. In L. WANNER, Ed., *Selected Lexical and Grammatical Issues in the Meaning-Text Theory. In Honour of Igor Mel'čuk*, volume 84 of *Language Companion Series*, p. 43–95. Amsterdam/Philadelphia : John Benjamins.

POLGUÈRE A. (2009). Lexical systems : graph models of natural language lexicons. *Language Resources and Evaluation*, **43**(1), 41–55.

POLGUÈRE A. (2011). Classification sémantique des lexies fondée sur le paraphrasage. *Cahiers de lexicologie*, **98**, 197–211.

POLGUÈRE A. (2012a). Lexicographie des dictionnaires virtuels. In Y. APRESJAN, I. BOGUSLAVSKY, M.-C. L'HOMME, L. IOMDIN, J. MILIĆEVIĆ, A. POLGUÈRE & L. WANNER, Eds., *Meanings, Texts, and Other Exciting Things. A Festschrift to Commemorate the 80th Anniversary of Professor Igor Alexandrovič Mel'čuk*, Studia Philologica, p. 509–523. Moscou : Jazyki slavjanskoj kultury Publishers.

POLGUÈRE A. (2012b). Like a Lexicographer Weaving Her Lexical Network. In *Proceedings of the Third Workshop on Cognitive Aspects of the Lexicon (CogALex III)*, p. 1–3, Mumbai : The COLING 2012 Organizing Committee.

POLGUÈRE A. (2013). Les petits soucis ne poussent plus dans le champ lexical des sentiments. In F. BAIDER & G. CISLARU, Eds., *Cartographie des émotions. Propositions linguistiques et sociolinguistiques*, p. 21–41. Paris : Presses Sorbonne Nouvelle.

POLGUÈRE A. & SIKORA D. (2013). Modèle lexicographique de croissance du vocabulaire fondé sur un processus aléatoire, mais systématique. In C. GARCIA-DEBANC, C. MASSERON & C. RONVEAUX, Eds., *Enseigner le lexique*, Recherches en didactique du français 5, p. 35–63. Namur : Presses Universitaires de Namur.

SAGOT B. & FIŠER D. (2008). Building a free French wordnet from multilingual resources. In *Proceedings of OntoLex 2008*, Marrakech.

SAJOUS F., NAVARRO E. & GAUME B. (2011). Enrichissement de lexiques sémantiques approvisionnés par les foules : le système WISIGOTH appliqué à Wiktionary. *Traitement Automatique des Langues (T.A.L.)*, **52**(1), 11–35.

SPOHR D. (2012). *Towards a Multifunctional Lexical Resource. Design and Implementation of a Graph-based Lexicon Model*. Berlin/Boston : De Gruyter.

WATTS D. J. & STROGATZ S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, **393**, 440–442.

WIERZBICKA A. (2009). The theory of the mental lexicon. In S. KEMPGEN, P. KOSTA, T. BERGER & K. GUTSCHMIDT, Eds., *Die slavischen Sprachen/The Slavic Languages : An International Handbook of their Structure, their History and their Investigation*, p. 848–863. Berlin & New York : Mouton de Gruyter.

WOLTER B. (2006). Lexical Network Structures and L2 Vocabulary Acquisition : The Role of L1 Lexical/Conceptual Knowledge. *Applied Linguistics*, **27**(4), 741–747.

ZOCK M. (2005). Le dictionnaire mental, modèle des dictionnaires de demain ? *Revue française de linguistique appliquée*, **X**(2), 103–117.

Un modèle pour prédire la complexité lexicale et graduer les mots

Núria Gala[1] Thomas François[2] Delphine Bernhard[3] Cédrick Fairon[2]
(1) LIF-CNRS UMR 7279, Aix Marseille Université,
(2) CENTAL, Université Catholique de Louvain,
(3) LILPA, Université de Strasbourg
nuria.gala@lif.univ-mrs.fr, tfrancois@uclouvain.be, dbernhard@unistra.fr, cfairon@uclouvain.be

Résumé. Analyser la complexité lexicale est une tâche qui, depuis toujours, a principalement retenu l'attention de psycholinguistes et d'enseignants de langues. Plus récemment, cette problématique a fait l'objet d'un intérêt grandissant dans le domaine du traitement automatique des langues (TAL) et, en particulier, en simplification automatique de textes. L'objectif de cette tâche est d'identifier des termes et des structures difficiles à comprendre par un public cible et de proposer des outils de simplification automatisée de ces contenus. Cet article aborde la question lexicale en identifiant un ensemble de prédicteurs de la complexité lexicale et en évaluant leur efficacité via une analyse corrélationnelle. Les meilleures de ces variables ont été intégrées dans un modèle capable de prédire la difficulté lexicale dans un contexte d'apprentissage du français.

Abstract. Analysing lexical complexity is a task that has mainly attracted the attention of psycholinguists and language teachers. More recently, this issue has seen a growing interest in the field of Natural Language Processing (NLP) and, in particular, that of automatic text simplification. The aim of this task is to identify words and structures which may be difficult to understand by a target audience and provide automated tools to simplify these contents. This article focuses on the lexical issue by identifying a set of predictors of the lexical complexity whose efficiency are assessed with a correlational analysis. The best of those variables are integrated into a model able to predict the difficulty of words for learners of French.

Mots-clés : complexité lexicale, analyse morphologique, mots gradués, ressources lexicales.

Keywords: lexical complexity, morphological analysis, graded words, lexical resources.

1 Introduction

La complexité lexicale n'est pas une notion qui puisse être définie dans l'absolu. En effet, un terme est perçu différemment en fonction du public qui y est confronté (apprenants de langue maternelle, apprenants de langue seconde, personnes avec une difficulté ou une pathologie liée au langage, etc.), d'où le terme de 'difficulté' (complexité subjective, (Blache, 2011)). De même, s'appuyer sur le seul critère de la fréquence pour appréhender la complexité du lexique semble réducteur : bien que ce critère se soit avéré très efficace dans la littérature (voir section 2), cette variable ne peut seule expliquer l'ensemble des problèmes rencontrés par différentes catégories de lecteurs. La notion de 'complexité' est, ainsi, multidimensionnelle (vitesse d'accès au lexique mental, compréhension, mémorisation, prononciation, activation du sens, orthographe, etc.), difficilement saisissable à partir de critères uniquement statistiques et très liée aux caractéristiques du public envisagé.

Dans le cadre de cet article, nous visons un public d'apprenants du français langue maternelle (L1) ou de français langue étrangère (FLE). En tenant compte de plusieurs ressources existantes, nous avons identifié un ensemble de variables intra-lexicales et statistiques que nous avons intégrées dans un modèle statistique cherchant à prédire le degré de complexité de mots dont la difficulté a été annotée par ailleurs. Notre hypothèse est que la combinaison de plusieurs variables intralexicales fines, associées à des informations statistiques, peut donner des indications plus précises sur le degré de complexité d'un mot. Dans ce sens, après un état de l'art introductif à la section 2, nous présentons la méthodologie et les ressources que nous avons utilisées pour identifier des variables susceptibles de caractériser la complexité lexicale (section 3). Dans un deuxième temps, nous présentons ces variables et nous discutons de leur impact à la section 4. À la section 5, nous décrivons le modèle de difficulté intégrant ces prédicteurs et nous analysons les résultats obtenus. Enfin, nous concluons l'article par une discussion sur notre approche et les résultats obtenus à la section 6, avant de proposer quelques futures améliorations à la section 7.

2 État de l'art

Analyser la complexité lexicale est une tâche qui, depuis toujours, a principalement intéressé les psycholinguistes et les pédagogues. En effet, de nombreux travaux sont décrits dans la littérature et se basent, par exemple, sur des tâches telles que la décision lexicale, la catégorisation sémantique, etc. pour explorer les propriétés du lexique. Ainsi, l'un des critères majeurs pour considérer qu'un mot est simple ou complexe est celui de la fréquence : de nombreux travaux démontrent la corrélation étroite entre la haute fréquence d'un terme et le fait que celui-ci soit perçu comme plus 'simple' (Howes & Solomon, 1951; Monsell, 1991). C'est d'ailleurs le critère que plusieurs auteurs avaient utilisé dans la première moitié du 20e siècle pour construire les premières ressources de lexique 'simplifié', par exemple la liste de Thorndike (1921), le *Teachers' Book of Words*, qui reprend les 20 000 mots les plus courants de la langue anglaise assortis de leur fréquence d'usage, ou encore le *Français fondamental* de Gougenheim (1958) qui comprend 1 500 mots usuels pour l'apprentissage du français, aussi bien en tant que langue étrangère que maternelle. La liste de Thorndike reste une référence dans le domaine de la lisibilité (avant l'apparition des listes obtenues par traitement informatisé). Elle s'avère un instrument de mesure objectif de la difficulté lexicale des textes et ce malgré quelques faiblesses, comme la mauvaise estimation des fréquences des mots appelés *disponibles* (mots avec fréquence variée selon les corpus mais usuels et utiles [1]).

D'autres critères avancés dans la littérature pour identifier des mots 'simples' concernent plutôt la familiarité d'un terme (Gernsbacher, 1984) ou encore son âge d'acquisition (Brysbaert *et al.*, 2000). La familiarité lexicale a été utilisée pour la constitution d'une liste de mots simples par Dale (1931). Dans l'expérience menée par Dale et ses collègues, la mesure de familiarité a été définie comme suit : dans une liste de 10 000 mots, n'ont été retenus que les termes connus par au moins 80% des élèves de quatrième primaire (CM1), ce qui a réduit la liste à 3 000 mots. Le nombre de voisins orthographiques (nombre d'unités de même longueur ne se différenciant que par une seule lettre) a aussi été envisagé par Coltheart *et al.* (1977) comme une mesure discriminante de la difficulté d'accès au lexique mental, même si les résultats dans des tâches de décision lexicale semblent varier selon les langues. Enfin, la longueur (en nombre de syllabes et/ou caractères) apparaît aussi comme un facteur déterminant dans la façon de percevoir les unités lexicales, en particulier parce qu'un mot plus long augmente la probabilité de fixer la fovéa (zone de la rétine où la vision des détails est la plus précise) sur un point de position non optimal, ce qui engendre une perte de temps à la lecture (Vitu *et al.*, 1990). Plus récemment, Schreuder & Baayen (1997) démontrent que le nombre de morphèmes et la taille de la famille morphologique jouent un rôle dans la décision lexicale visuelle (reconnaissance de mots parmi une série de mots et non-mots). Laufer (1997), pour sa part, identifie une série de facteurs linguistiques influençant l'acquisition du lexique, parmi lesquels la familiarité des phonèmes, la régularité dans la prononciation, la cohérence graphème-phonème, la transparence morphologique ou la polysémie. Potentiellement, ces facteurs contribuent tous à la façon dont les mots sont perçus.

Les répercussions de ces travaux sont d'abord théoriques, aidant, par exemple, à comprendre l'organisation du lexique mental et comment il se distribue dans les différentes zones du cerveau. D'un point de vue plus pratique, certaines de ces études ont cependant débouché sur la construction de listes utilisées pour l'enseignement des langues. Plus récemment, la question de l'évaluation de la difficulté lexicale a fait l'objet d'un intérêt grandissant dans le domaine du traitement automatique des langues (TAL) et, en particulier, en simplification automatique de textes. Dans ce domaine, le but reste d'identifier des termes et des structures difficiles à comprendre par un public cible et de proposer des outils de simplification automatisée de ces contenus. Bien que la plupart des travaux en simplification de textes se focalisent sur des aspects syntaxiques (par exemple (Chandrasekar *et al.*, 1996)), certains auteurs ont mis en oeuvre des systèmes qui visent le traitement du lexique. Dans ce cas, différents aspects doivent être pris en compte : (i) la détection des mots ou termes complexes à remplacer, (ii) l'identification de substituts et (iii) l'adéquation au contexte. Ces trois aspects ne sont pas toujours pris en compte de manière conjointe. Sous sa forme la plus simple, la substitution lexicale se fait en fonction de la fréquence des synonymes extraits d'une ressource comme WordNet, sans prise en compte du contexte (Carroll *et al.*, 1998). Récemment, des travaux ont fait appel à des corpus comparables comme Wikipedia et sa version simplifiée pour l'anglais (*Simple English Wikipedia*) pour acquérir des ressources utiles pour la simplification lexicale : ainsi, Biran *et al.* (2011) proposent une mesure de la complexité d'un mot qui est fonction de sa fréquence dans les deux versions de Wikipedia et de sa longueur. D'une manière générale, les critères utilisés pour sélectionner le meilleur substitut restent relativement simples. Pour la tâche de simplification lexicale organisée lors de la campagne SemEval 2012 (Specia *et al.*, 2012), la *baseline* correspondant à une simple mesure de fréquence dans un gros corpus n'a été battue que par un seul système. Ce résultat rend compte de la difficulté de la tâche : même si les travaux en psycholinguistique ont mis en évidence des facteurs complexes, leur intégration dans des systèmes automatisés n'est pas encore résolue.

1. Par exemple "fourchette", "coude", etc.

3 Exploitation de ressources existantes

Pour réaliser les différentes expériences présentées dans cet article, nous avons eu recours à un ensemble de ressources qui ont été utilisées en vue de deux objectifs : certaines ressources lexicales ont servi de liste de référence pour l'apprentissage du modèle, tandis que les autres ressources ont été employées pour récupérer diverses informations linguistiques utilisées dans nos variables.

3.1 Ressources d'apprentissage

Pour entraîner un modèle statistique capable de prédire ou comparer la difficulté de mots, il importe de disposer d'un nombre suffisant de mots dont la difficulté est connue et exprimée en fonction d'une unité pratique. En psycholinguistique, il est commun d'associer le temps de réponse nécessaire pour réaliser une tâche associée à un mot à la difficulté de ce mot (Ferrand, 2007). Cependant, cette approche nécessite de disposer d'un nombre important de sujets et de moyens, en particulier lorsqu'on envisage un large vocabulaire. C'est pourquoi nous avons choisi de constituer notre ressource d'entraînement d'une autre façon : en nous basant sur l'association des mots à des niveaux scolaires déterminés, calculés sur la base de l'apparition de ces mots dans des manuels scolaires. Notre hypothèse est qu'un mot facile apparaîtra en général plus tôt dans les manuels scolaires qu'un mot plus complexe. Par chance, il existe deux ressources pour le français qui recensent les mots utilisés dans des manuels scolaires de différents niveaux, à savoir Manulex (Lété *et al.*, 2004) et FLELex (François *et al.*, 2014).

Manulex [2] a été créée à partir de 54 manuels scolaires (pour un total de 1,9 millions d'occurrences). Il décrit la distribution d'unités lexicales en fonction de leur apparition dans des manuels qui ont été classés en trois niveaux : (1) la première année de primaire ou CP (6 ans), (2) la deuxième année ou CE1 (7 ans) et (3) une catégorie qui regroupe les trois années suivantes (CE2-CM2, 8-11 ans). Ce choix se justifie en termes de volume d'acquisition de vocabulaire : au CP, se construit le lexique de l'enfant sur la base de la médiation phonologique ; au CE1, se construit le lexique orthographique par automatisation progressive de la reconnaissance du mot écrit et au cycle 3, le stock lexical se consolide et s'enrichit par exposition répétée à l'écrit [3]. La ressource, librement disponible, totalise 23 812 lemmes, mais nous n'avons conservés que les mots lexicaux (noms, adjectifs, adverbes et verbes), ce qui réduit le nombre de lemmes à 19 038. Il faut aussi signaler que les fréquences associées à chaque mot de la ressource ne correspondent pas aux valeurs absolues observées dans les manuels, mais à des valeurs adaptées en fonction d'un indice de dispersion qui augmente l'importance des termes en fonction du nombre de documents dans lesquels ils sont apparus. La Figure 1 présente un exemple d'entrées issues de Manulex.

lemme	POS	Fréq. N1	Fréq N2	Fréq. N3
pomme	N	724	306	224
vieillard	N	-	13	68
patriarche	N	-	-	1
cambrioleur	N	2	-	33

TABLE 1 – Exemple d'entrées de Manulex

FLELex, quant à lui, a été obtenu à l'aide d'une méthodologie similaire, mais sur la base d'un corpus de 28 manuels de français langue étrangère (FLE) et de 29 livres simplifiés également destinés à des lecteurs en FLE. Ces ouvrages étaient classés selon l'échelle de difficulté proposée par le cadre européen commun de référence pour les langues (Conseil de l'Europe, 2001) ou CECR, qui définit six niveaux de maîtrise communicationnelle : A1 (niveau introductif ou de survie) ; A2 (niveau intermédiaire) ; B1 (niveau seuil) ; B2 (niveau avancé ou utilisateur indépendant) ; C1 (niveau autonome ou de compétence opérationnelle effective) et C2 (maîtrise). La ressource totalise 14 053 lemmes lexicaux et 183 lemmes grammaticaux, dont les fréquences ont été estimées sur 777 835 occurrences.

Dans les deux cas, et comme le montre la Table 1, le problème de ces ressources par rapport à notre propre objectif de recherche est qu'elles offrent la distribution des fréquences de chaque mot par niveau, mais n'associent pas strictement un mot à un niveau donné. C'est pourquoi nous avons dû transformer ces distributions en un niveau. Trois techniques ont été testées pour ce faire. La plus simple d'entre elles consiste à attribuer à un mot le premier niveau où il a été observé dans le corpus. Ainsi pour Manulex, *pomme* et *cambrioleur* se voient attribuer le niveau 1, tandis que *patriarche* est associé au

2. http://www.manulex.org
3. http://leadserv.u-bourgogne.fr/bases/manulex/manulexbase/indexFR.htm

niveau 3. On comprend aisément que cette façon de faire, qui assimile la distribution de *pomme* à celle de *cambrioleur* n'est pas optimale. C'est pourquoi nous avons également considéré chaque distribution comme une série statistique (ex. pour *pomme*, une série constituée de 724 un, de 306 deux et de 224 trois, les chiffres 'un' à 'trois' renvoyant respectivement aux trois niveaux scolaires, voir Table 1) et pris comme valeur représentative soit son premier quartile, soit sa moyenne (ce qui donne alors une échelle continue, comprise entre 1 et 3 pour Manulex et 1 et 6 pour FLELex).

3.2 Ressources pour l'extraction de variables

Pour l'extraction de variables, nous avons mobilisé plusieurs ressources contenant différentes informations linguistiques. Leur usage particulier au sein de nos variables est décrit plus en détail dans la section 4, cette section présentant ces ressources de façon plus générale.

La première d'entre elles est Lexique 3 [4] (New *et al.*, 2001). il s'agit d'une ressource librement accessible qui contient un grand nombre d'informations linguistiques (transcription phonétique, structure syllabique, flexion, etc.) et statistiques (nombre de phonèmes, de syllabes, de morphèmes, fréquence dans des corpus de livres et de sous-titre de films, etc.). Elle contient 142 728 mots correspondant à 47 342 lemmes.

Polymots [5] (Gala & Rey, 2008) est un lexique morphologique. Les mots ont été segmentés morphologiquement en bases et affixes, des informations sur les familles morphologiques et sur des unités de sens associées sont également disponibles. La version 3 contient 19 510 lemmes et 2 364 familles. La segmentation morphologique et le regroupement en familles ont été effectués manuellement, ce qui a comme répercussion une couverture assez faible (par rapport à d'autres ressources comme Lexique3, par exemple). Ainsi, l'intersection entre Manulex (restreint aux mots lexicaux) et Polymots est de 55,75 %, c'est-à-dire qu'il y a 10 614 mots communs entre les deux ressources [6]. De ce fait, nous avons décidé d'utiliser une méthode par apprentissage non supervisé pour l'obtention des variables morphologiques. Les lemmes de Polymots, tout comme ceux de Morphalou 2.0 (Romary *et al.*, 2004), nous ont servi à enrichir notre corpus d'apprentissage pour l'analyse morphologique.

Nous avons également utilisé un corpus issu d'enregistrements de patients atteints de la maladie de Parkinson (2 271 formes pour 373 lemmes). Il s'agit d'une vingtaine d'enregistrements (correspondant à une tâche de description d'une image de la vie quotidienne) de patients en état "off", c'est-à-dire, sans médicaments qui pourraient inhiber les effets de la maladie sur la parole. Nous nous sommes intéressés à ce type de parole pathologique car la maladie de Parkinson, bien qu'elle soit plus connue pour des symptômes moteurs (tremblements, rigidité musculaire, etc.), entraîne également des difficultés au niveau de la parole (Pinto *et al.*, 2010). Par conséquent, nous postulons que ce type de parole pathologique peut être représentative d'une langue plus simple et donc d'un lexique plus simple. La classification que nous proposons des structures syllabiques (variable 8, section 4) est issue d'observations faites sur ce corpus. Les données de ce corpus ont aussi servi à enrichir le corpus d'apprentissage pour l'extraction des informations morphologiques.

Enfin, pour l'obtention de variables sémantiques, nous avons utilisé les réseaux lexicaux JeuxDeMots [7] (Lafourcade, 2007) et BabelNet (Navigli & Ponzetto, 2010). JeuxdeMots contient à ce jour 314 494 termes, dont 136 421 ont au moins une relation de type idée associée (synonymie, hyperonymie, etc.). Des 19 037 lemmes de Manulex, 6 068 sont étiquetés comme polysémiques (31,2%). Nous avons utilisé cette ressource pour extraire des synonymes pour les mots de Manulex. Quant à BabelNet, il s'agit d'un réseau multilingue construit à partir de WordNet et Wikipédia. Nous avons utilisé les informations sur les 23 242 lemmes du français, en particulier le nombre de synsets associés.

4 Analyse de variables pour caractériser la complexité lexicale

4.1 Typologie

Dans cette section nous introduisons un ensemble de variables présentées ci-dessous. Nous mettons l'accent (en gras) sur les variables morphologiques et sémantiques, qui constituent les deux apports principaux de cet article par rapport à

4. http://www.lexique.org/
5. http://polymots.lif.univ-mrs.fr
6. À la base, Polymots ne contient pas des mots composés ('mainmise'), ni des noms avec tiret ('amour-propre'), des mots originaires d'autres langues ('mortadelle'), des mots grammaticaux ('tellement') ou encore des mots techniques ('dyoxide'). La création manuelle ainsi que l'écart de ces mots justifient sa faible couverture par rapport à Manulex
7. http://www.jeuxdemots.org/

des approches proches dans ce domaine (Gala *et al.*, 2013). C'est pourquoi, la section 4.2 est consacrée à la description détaillée de l'implémentation des variables morphologiques, les autres étant directement décrites dans la liste ci-dessous.

4.1.1 Critères orthographiques

1. *Nombre de lettres* : nombre de caractères alphabétiques dans un mot ;
2. *Nombre de phonèmes* : pour calculer le nombre de phonèmes dans un mot, un système mixte a été mis en place. Pour les mots présents dans Lexique3, nous avons simplement récupéré l'information issue de cette ressource. Pour les mots absents de Lexique3, nous avons généré leur représentation phonétique au vol via *eSpeak* [8] ;
3. *Nombre de syllabes* : comme pour le nombre de phonèmes, le nombre de syllabes d'un mot a soit été récupéré directement dans Lexique3, quand l'information était disponible, soit a été calculé automatiquement en deux étapes. Tout d'abord, la forme phonétique a été générée (comme au point précédent), avant d'y appliquer l'outil de syllabification de Pallier (1999) ;
4. *Voisinage orthographique* : les informations concernant le nombre ou la fréquence des voisins orthographiques [9] proviennent également de Lexique 3 et nous les avons déclinées en 3 variables : (4a) nombre de voisins, (4b) fréquence cumulée de tous les voisins, (4c) nombre des voisins les plus fréquents ;
5. *Cohérence phonème-graphie* : le nombre de phonèmes et de lettres dans un mot ont été comparés sur la base de la classification suivante : 0 pour l'absence de différence (c'est-à-dire, une transparence parfaite), par exemple *abruti* [abRyti] ; 1 pour une différence de 1 ou 2 caractères, par exemple *abriter* [abRite] ; 2 pour une différence supérieure à 2 caractères, par exemple dans *lentement* [l@t-m@] [10] ;
6. *Patrons orthographiques* : 5 variables ont été définies autour de la présence de graphèmes complexes dans les mots, à savoir (6a) des voyelles orales (par ex. 'au' [o]), (6b) des voyelles nasales (par ex ; 'in' [ɛ̃]), (6c) des doubles consonnes (par ex. 'pp'), (6d) des doubles voyelles (par ex. 'éé'), (6e) ou encore des digrammes (par ex. 'ch' [ʃ]) ;
7. *Structure syllabique* : trois niveaux de complexité pour les structures syllabiques présentes dans les mots ont été définis sur la base des fréquences de ces structures dans le corpus de parole « simple » Parkinson : (7a) les structures les plus fréquentes [11] (CYV, V, CVC, CV), (7b) les structures relativement fréquentes (CCVC, VCC, VC, YV, CVY, CYVC, CVCC, CCV), (7c) et les structures peu fréquentes (combinaisons de plusieurs consonnes, par exemple CCCVC) ;

4.1.2 Critères morphologiques

8. ***Nombre de morphèmes*** : nombre total de préfixes, suffixes et de bases dans le mot ;
9. ***Fréquence minimale des affixes (préfixes et suffixes)*** : nombre de mots différents (types) dans lesquels apparaît le préfixe / suffixe le moins fréquent ;
10. ***Fréquence moyenne des affixes (préfixes et suffixes)*** : moyenne des fréquences absolues des préfixes / suffixes ;
11. ***Préfixation*** : attestation ou non de la présence de préfixes ;
12. ***Suffixation*** : attestation ou non de la présence de suffixes ;
13. ***Composition*** : attestation ou non de la présence de deux bases ou plus ;
14. ***Taille de la famille morphologique*** : voir section 4.2 ;

4.1.3 Critères sémantiques

15. *Polysémie selon JeuxdeMots* : booléen indiquant si le mot est polysémique ou non ;
16. *Polysémie selon BabelNet* : nombre de synsets répertoriés dans BabelNet ;

8. http ://espeak.sourceforge.net
9. Les voisins orthographiques regroupent l'ensemble des mots de même longueur ne se différenciant que par une seule lettre tels que, pour SAGE, les mots MAGE, SALE, etc.).
10. La transcription est celle de Lexique 3 qui utilise l'alphabet SAMPA (*Speech Assessment Methods Phonetic Alphabet*).
11. La notation utilisée est la suivante : C pour consonne, V pour voyelle, Y pour les glides [j], [w] et [ɥ].

4.1.4 Critères statistiques

17. *Fréquence dans Lexique3* : logarithme des fréquences extraites de Lexique3 (calculées à partir d'un corpus de sous-titres de films). Pour traiter les mots absents de la ressource, nous avons appliqué un algorithme de lissage par Good-Turing (Gale & Sampson, 1995) afin d'attribuer une log-probabilité très petite par défaut à ces termes hors vocabulaire ;
18. *Présence/absence dans la liste de Gougenheim* : pour chaque mot, un booléen indique s'il appartient ou non à la liste du *Français Fondamental* dans sa version longue (qui comprend 8 875 lemmes). Comme il est bien connu en lisibilité que la taille de la liste de mots simples utilisée comme variable influe sur la capacité de discrimination de celle-ci, nous avons expérimenté avec diverses tailles de liste, par tranche de 1 000 mots, de 1 000 à 8 875 mots.

4.2 Obtention des variables par analyse morphologique

Les variables morphologiques 8 à 14 ont été obtenues par analyse morphologique non supervisée, en utilisant les systèmes décrits dans (Bernhard, 2006) et (Bernhard, 2010). Pour ce faire, les lemmes issus de Morphalou 2.0, Manulex, Polymots, corpus Parkinson et FLELex ont été fusionnés et ont été associés à leur fréquence dans le corpus `2010-wiki-1M` [12] du portail Wortschatz (Quasthoff *et al.*, 2006), qui contient 1 million de phrases issues de Wikipédia. Cette liste de lemmes associés à leur fréquence sert à l'apprentissage non supervisé d'informations morphologiques.

Le premier système (Bernhard, 2006) permet le découpage des mots en segments morphémiques étiquetés : base (b), préfixe (p), suffixe (s) et élément de liaison (l), comme par exemple `im_p + pens_b + able_s + ment_s`. La première étape du système consiste à extraire une liste de préfixes et de suffixes sur la base des probabilités transitionnelles entre sous-chaînes observées dans le lexique. Cette étape est contrôlée par un paramètre N qui détermine la quantité de préfixes et suffixes extraits : dans nos expériences, nous avons fixé ce paramètre à 5 et 10, pour tester son influence sur les variables utilisées dans le modèle final. La deuxième étape permet d'extraire une liste de bases candidates en utilisant les affixes obtenus à l'étape 1. Les deux dernières étapes consistent à segmenter les mots et à identifier la meilleure segmentation possible.

Ce système a été utilisé pour définir la valeur des variables suivantes : nombre de morphèmes, préfixation (oui/non), suffixation (oui/non), est composé (oui/non), fréquence minimale des préfixes, fréquence minimale des suffixes, fréquence moyenne des préfixes, fréquence moyenne des suffixes. Un mot est considéré comme composé s'il contient deux bases dans la segmentation.

Les résultats de ce premier système ont également permis de déterminer la taille de la famille morphologique : tous les mots qui contiennent la même base sont regroupés dans la même famille. Ainsi, la famille d'« impensablement » contient tous les mots contenant le segment `pens_b`. Pour les mots composés qui appartiennent à plusieurs familles, seule la taille de la famille la plus petite a été prise en compte.

Les deux derniers systèmes, MorphoClust et MorphoNet (Bernhard, 2010) produisent uniquement des familles morphologiques et n'ont donc été utilisés que pour obtenir la taille de la famille. MorphoClust forme des familles par classification ascendante hiérarchique. La méthode MorphoNet est quant à elle fondée sur la détection de communautés dans des réseaux lexicaux. MorphoClust utilise la même première étape que le système de segmentation et est donc dépendant du même paramètre N, encore une fois fixé à 5 et 10. Des expérimentations conduites précédemment ont montré que MorphoClust obtient un meilleur rappel, tandis que MorphoNet est plus précis (Bernhard, 2010).

5 Intégration dans un modèle pour prédire la difficulté

Après avoir identifié un ensemble de 49 variables lexicales (les 18 mentionnées ci-dessus et leurs variantes, c'est-à-dire, par exemple, la proportion d'absents calculée sur la base de listes de taille différentes), nous avons effectué deux séries d'expériences à partir de nos deux listes de mots gradués (Manulex et FLELex). Dans un premier temps, l'efficacité de ces variables a été évaluée par une analyse bivariée, c'est-à-dire sans tenir compte de la multicolinéarité [13] présente au sein des données. L'objectif de cette étape est simplement de déterminer quels sont, parmi nos prédicteurs, ceux qui apportent le

12. Pour les mots absents du corpus, la fréquence a été fixée à 1.
13. Il s'agit de la redondance informationelle partielle entre certaines variables. Par exemple, les mots plus courts sont aussi souvent les plus polysémiques et les plus fréquents.

plus d'information sur la difficulté des mots. Dans un second temps, l'ensemble des variables significativement associées à la difficulté lexicale ont été combinées au sein d'un algorithme d'apprentissage automatisé (SVM).

5.1 Analyse des variables

Comme nous l'avons rappelé aux sections 1 et 4, les variables sélectionnées pour cette étude l'ont été sur la base de résultats en psycholinguistique attestant d'un lien de causalité entre ces caractéristiques lexicales et la difficulté des mots mesurées au moyen de multiples tâches. De plus, (Gala *et al.*, 2013) ont analysé le comportement d'un sous-ensemble de nos variables en lien avec la parole "pathologique" via le corpus Parkinson décrit à la section 3.2. Cette étude a comparé les caractéristiques des mots utilisés dans un langage supposé plus "simple" avec un lexique général (Lexique3). Il est apparu que les unités lexicales "simples" étaient effectivement plus courtes (6,3 lettres, 4,7 phonèmes, 1,96 syllabes contre 8,6 lettres, 6,8 phonèmes et 2,89 syllabes dans Lexique3). Les structures syllabiques des ces mots, étaient également plus simples. Cela est une première piste qui confirmerait, sur un large ensemble de données réelles, la relation existante entre certaines de nos variables et la difficulté des mots.

Pour obtenir un diagnostic plus précis de l'efficacité de chaque variable, une analyse corrélationnelle a été effectuée. Pour chaque variable, la force de sa relation avec la difficulté lexicale a été mesurée à l'aide du coefficient de corrélation de Spearman [14]. Nous avons répété cette analyse sur nos deux lexiques de référence : Manulex et FLELex. Comme ces ressources ne comprennent pas les mêmes mots et ceux-ci ne sont pas classés de la même façon, cela devrait augmenter la robustesse de nos conclusions quant à l'efficacité de nos variables comme prédicteurs de la difficulté lexicale en général.

Le résumé des analyses corrélationnelles est présenté à la Table 2. Les niveaux utilisés pour cette analyse sont ceux obtenus via la première des trois techniques de transformation de la distribution d'un mot en un niveau (voir section 3.1), à savoir le premier niveau où apparaît ce mot. L'analyse corrélationnelle a également été effectuée pour les autres techniques, mais les corrélations obtenues étaient inférieures. Par exemple, pour Manulex, la corrélation entre la log-fréquence des mots dans Lexique3 et la difficulté tombe à $\rho = -0,40$ pour la moyenne et à $-0,36$ pour le quartile. Il semble donc que la meilleure mesure de la difficulté d'un mot parmi celles que nous avons testées sur la base d'une liste de mots gradués, soit la première apparition d'un mot dans un niveau donné.

Variables	**Manulex (ρ)**	**FLELex (ρ)**
17 Freq. Lex3	-0,51	-0.53
18 AbsGoug (5000)	-0,41	-0.46
18 AbsGoug (4000)	-0,41	-0.47
02 Nb. phon	0,30	0,27
15 Polysémie	-0,29	-0.38
01 Nb. lettres	0,27	0,25
03 Nb. syllables	0,27	0,26
4a Nb. voisin	-0,25	-0,23
4b Voisin freqcum	-0,25	-0,23
16 Synset BabelNet	-0,20	-0,19
6b Voy. Nasale	0,08	0,07
14 Taille famille (morphoclust_10)	-0,08	-0,05
11 Prefix (seg_10)	0.07	0,06
08 Nb_morphs (seg_10)	0,06	0,08
06 Patrons ortho (a-d)	0,05	0,06
10 Moyenne sufx. (freq_seg_10)	-0,05	0,02

TABLE 2 – Sélection des meilleurs variables

Les résultats de cette analyse corrélationnelle semblent assez clairs et stables. En effet, les corrélations obtenues sur Manulex sont très proches de celles obtenues sur FLELex. S'il est vrai qu'une partie non négligeable de mots se retrouvent dans les deux listes (mêmes observations), ceux-ci n'ont pas nécessairement été annotés de la même manière. On note que les meilleures variables sont celles basées sur la fréquence (17) et sur la présence dans une liste de mots simples (18) de taille moyenne (entre 4 000 et 5 000 mots). Il est intéressant de noter que l'appartenance des mots à une liste de mots

14. Le choix de privilégier ce coefficient par rapport au coefficient de Pearson, plus commun, est essentiellement motivé par le fait que la relation entre les variables linguistiques et la difficulté n'est pas nécessairement linéaire, comme discuté dans François (2011). La corrélation de Spearman est plus adaptée pour capturer des relations monotones croissantes.

simples est plus discriminant en FLE que pour le français L1, de même qu'une liste plus courte est préférable pour le FLE, comme cela avait été montré par François (2011).

Une seconde constatation d'intérêt est l'efficacité et la robustesse des variables classiques telles que le nombre de lettres (01) et de syllabes (03). Moins utilisé dans la littérature, le nombre de phonèmes dans un mot (02) apparaît tout aussi efficace. Enfin, les informations relatives au statut polysémique des mots apportent également de l'information qui semble utile pour expliquer la difficulté du lexique, que ce soit via une information binaire sur le statut polysémique des mots issue de JeuxDeMots (15) ou via le nombre de synsets repris dans BabelNet (16). À notre connaissance, la question de la polysémie n'a pas encore été prise en compte en lisibilité et sa performance constitue une bonne surprise, d'autant que l'information véhiculée par cette variable ne recouvre que faiblement l'effet de fréquence [15].

Enfin, d'autres variables sont également significativement corrélées à la difficulté, mais dans une moindre mesure. C'est le cas des informations concernant les voisins orthographiques (4a, 4b), les patrons orthographiques complexes (06), en particulier les voyelles nasales (6b), ou encore des informations morphologiques. A ce niveau, c'est surtout la taille de la famille (14), le nombre de morphèmes (08) et la présence d'un préfixe (11) qui apparaissent les plus utiles parmi l'ensemble de nos variables morphologiques. Signalons que la taille de la famille morphologique apparaissait déjà comme une variable significative pour (Schreuder & Baayen, 1997) dans une tâche de décision lexicale : plus la famille morphologique est grande, plus cela tend à induire un effet de facilitation. Toutefois, même si ces corrélations sont largement significatives (en terme de p-valeur) au vu du nombre élevé de données [16], on peut conclure que l'effet des variables morphologiques reste assez faible pour identifier la complexité d'un mot.

5.2 Un modèle pour prédire la difficulté du lexique

Au terme de l'analyse corrélationnelle, une sélection a été effectuée parmi nos 49 variables sur la base de deux critères. D'une part, les variables retenues pour la phase de modélisation devaient être significativement corrélées à la difficulté. D'autre part, lorsqu'un ensemble de prédicteurs constituaient des variantes d'une même information déterminée par un paramètre (par exemple, la taille de la liste de Gougenheim ou le paramètre N pour les variables morphologiques), seul celui présentant la corrélation la plus élevée a été retenu. Cela nous donne un total de 26 variables pour Manulex et de 24 variables pour FLELex.

Sur la base de ces deux ensembles de variables, deux modèles statistiques ont été entraînés, l'un en se basant sur les mots de Manulex et l'autre sur ceux de FLELex. Pour rappel, les annotations utilisées pour cet entraînement l'ont été au moyen de notre première méthode : le niveau de la première occurrence d'un mot. Cela nous donne pour Manulex : 5 863 lemmes pour le niveau CP, 4 023 lemmes pour le CE1 et 9 151 lemmes pour le cycle 3. En ce qui concerne FLELex, la répartition par niveau est la suivante : 4 142 lemmes pour le niveau A1, 2 735 pour A2, 4 002 pour B1, 1 312 pour B2, 1 672 pour C1 et seulement 501 lemmes spécifiques au niveau C2.

L'algorithme d'apprentissage automatisé utilisé est une machine à vecteurs de support ou SVM (Boser *et al.*, 1992). Lors d'expériences préliminaires, nous avons utilisé la librairie LibSVM (Chang & Lin, 2011) et avons comparé différents kernels (linéaire, RBF et polynomial). Les résultats étant relativement similaires, nous avons finalement privilégié le recours à LibLinear (Fan *et al.*, 2008), qui même si elle ne permet que d'utiliser un kernel linéaire, s'est révélée considérablement plus rapide sur nos données. Afin de limiter les effets de multicolinéarité et de surapprentissage au sein des données, nous avons centré et réduit les données et opté pour une méthode de régularisation "L2". Enfin, le méta-paramètre C (le coût) a été choisi par une exploration limitée de l'espace de valeurs entre 100 et $0,001$.

Les résultats obtenus par les deux modèles sont repris au Tableau 3. Leurs performances sont évaluées en terme d'exactitude des prédictions du modèle sur des données de test, une valeur qui a été estimée à l'aide d'un algorithme de validation croisée à cinq échantillons. Les résultats sont comparés à deux baselines : (1) le score qu'obtiendrait un modèle prédisant toujours la classe majoritaire dans les données et (2) un modèle qui ne se baserait que sur l'information de fréquence, qui est la caractéristique lexicale la plus communément associée à la difficulté des mots.

On peut noter que nos deux modèles se comportent nettement mieux qu'un modèle qui attribuerait toujours la classe majoritaire. Pour Manulex, le gain par rapport à cette méthode simpliste est de 15% d'exactitude, tandis qu'il est de 14% pour FLELex. Le gain est donc relativement similaire dans les deux cas, même si les performances absolues ne sont pas équivalentes eu égard à la différence de classes entre les deux listes. En effet, effectuer des prédictions parmi six classes constitue une tâche plus complexe que prédire parmi trois catégories. Par contre le gain de performance par rapport à

15. Pour Manulex, la corrélation entre le statut polysémique des mots d'après JeuxDeMots (15) et la log-fréquence (17) n'était que de $\rho = -0,19$.
16. Par exemple, le nombre de morphèmes présente un ρ de $0,06$, ce qui correspond encore à un $p < 0,001$.

Liste	Modèle	C	Exac.	Ecart-type
Manulex	Classe majoritaire	/	48%	/
	Baseline Fréq.	0,1	61%	0,4%
	Modèle	0,5	63%	0,7%
FLELex	Classe majoritaire	/	28,8%	/
	Baseline Fréq.	0,5	39%	0,8%
	Modèle	0,001	43%	0,5%

TABLE 3 – Performances des modèles sur les deux listes de mots

la seconde baseline, qui n'utilise que l'information fréquentielle, est faible : 2% pour Manulex et 4% pour FLELex. Ce résultat est relativement surprenant en regard du nombre de variables dans le modèle, mais semble en accord avec la situation observée dans le cadre de SemEval 2012 où, rappelons-le, un seul modèle avait réussi à battre la baseline fréquentielle.

6 Discussion

Afin de mieux appréhender les résultats obtenus, nous avons effectué une série d'expérimentations supplémentaires sur les données. Tout d'abord, cette tendance des performances à plafonner semble être partiellement liée au déséquilibre entre les différentes classes au sein de nos deux jeux de données. Nous avons donc effectué des expériences complémentaires afin de vérifier l'impact de la distribution des classes sur le résultat de l'apprentissage. Ces expériences ont été réalisées avec *Weka 3.7.10* [17] avec un modèle de régression logistique (*SimpleLogistic*) en utilisant une validation croisée à dix échantillons. Deux méthodes de ré-échantillonage ont été comparées. La première consiste à obtenir une distribution uniforme des classes, en réduisant le nombre d'instances de manière à obtenir un nombre d'instance par classes à peu près égal au nombre d'instances de la classe la moins représentée. La seconde réduit les données d'apprentissage au même nombre d'instances que la première, mais en conservant une distribution équivalente aux données initiales [18]. L'intérêt de ce second processus est de contrôler l'effet de la taille du jeu de données sur les performances, afin de mieux isoler l'effet lié au type de distribution.

Les résultats de ces expériences sont détaillés dans les Tables 4 et 5. Le modèle de régression logistique obtient quasiment les mêmes résultats que ceux obtenus avec SVM. Par ailleurs, les résultats obtenus pour les données échantillonnées avec la même distribution sont comparables à ceux obtenus sur les données complètes. Toutefois, le ré-échantillonnage avec une distribution uniforme des classes conduit à des résultats largement inférieurs, alors même que le nombre d'instances est identique à celui du ré-échantillonage avec la même distribution que les données initiales.

	Données complètes (19 037 instances)	Données échantillonnées (11 993 instances)	
		Même distribution	Distribution uniforme
Fréquence	61,2%	61,6%	51,2%
Attributs sélectionnés	62,7%	63,0%	53,7%
Tous les attributs	62,9%	63,0%	53,4%

TABLE 4 – Exactitude pour la régression logistique sur le corpus Manulex, avec et sans ré-échantillonage

	Données complètes (14 364 instances)	Données échantillonnées (2 872 instances)	
		Même distribution	Distribution uniforme
Fréquence	41,1%	40,8%	28,6%
Attributs sélectionnés	42,7%	42,4%	33,2%
Tous les attributs	43,1%	41,9%	32,6%

TABLE 5 – Exactitude pour la régression logistique sur le corpus FLELex, avec et sans ré-échantillonage

17. http://www.cs.waikato.ac.nz/ml/weka/index.html
18. Le ré-échantillonage a été réalisé avec le filtre *Resample* disponible dans Weka, avec un tirage sans remise.

Pour expliquer ces différences, il est nécessaire d'étudier les performances obtenues indépendamment pour chaque classe. Ces résultats, exprimés cette fois à l'aide de la F-mesure, sont présentés dans les Tables 6 et 7 pour le modèle utilisant les attributs sélectionnés. On constate à nouveau des F-mesures comparables sur les données complètes et sur les données ré-échantillonnées avec une distribution similaire. Dans les deux cas, les F-mesures sont nulles ou proches de zéro pour certaines classes, tandis qu'elles s'élèvent pour les classes les plus représentées dans les données d'apprentissage. Les résultats obtenus pour une distribution uniforme sont plus équilibrés, mais montrent également une tendance intéressante : les classes situées aux deux extrémités des échelles sont celles pour lesquelles les résultats sont les meilleurs. Ceci impliquerait que les mots très simples ou très complexes sont plus faciles à identifier que les mots situés au milieu de l'échelle. Une tendance similaire a déjà été signalée au niveau du texte dans des travaux précédents en lisibilité (François & Fairon, 2012).

	Données complètes (19 037 instances)	**Données échantillonnées (11 993 instances)**	
		Même distribution	**Distribution uniforme**
CP (5 863)	0,644	0,648	0,636
CE1 (4 023)	0,000	0,000	0,355
Cycle 3 (9 151)	0,731	0,734	0,588

TABLE 6 – F-mesure par classe avec les attributs sélectionnés pour le corpus Manulex, avec et sans ré-échantillonage.

	Données complètes (14 364 instances)	**Données échantillonnées (2 872 instances)**	
		Même distribution	**Distribution uniforme**
A1 (4 142)	0,628	0,631	0,572
A2 (2 735)	0,029	0,041	0,326
B1 (4 002)	0,492	0,493	0,123
B2 (1 312)	0,000	0,023	0,251
C1 (1 672)	0,089	0,098	0,298
C2 (501)	0,000	0,000	0,352

TABLE 7 – F-mesure par classe avec les attributs sélectionnés pour le corpus FLELex, avec et sans ré-échantillonage.

7 Conclusions

Nous avons présenté un modèle de la complexité lexicale reposant sur diverses variables intralexicales et statistiques. L'un des aspects innovants de notre approche est que le degré de complexité des mots a été déterminé à partir de la présence de ces mots dans des manuels destinés à des apprenants du français langue maternelle et étrangère. Cette façon de faire nous a permis de traiter un large ensemble de termes, à la différence d'approches plus classiques en psycholinguistique qui associent la difficulté des mots à la vitesse de réalisation de tâches telles que la décision lexicale et qui demandent dès lors de soumettre chaque mot à un ensemble de sujets.

Un second point intéressant de l'étude est la prise en compte combinée d'un large ensemble de variables, toutes justifiées par des résultats de travaux issus du domaine de la psycholinguistique. Parmi ces variables, certaines n'ont que très peu été étudiées en lien avec la prédiction de la difficulté des textes ou des mots. C'est le cas des variables morphologiques, des types de structure syllabiques, des patrons orthographiques ou encore du statut polysémique des mots (d'après JeuxDeMots et BabelNet). Toutes nos variables ont été analysées afin de déterminer celles qui présentent la plus forte corrélation avec la difficulté lexicale telle que définie dans nos deux ressources (Manulex et FLELex). Certaines de ces variables telles que la fréquence et le nombre de caractères sont bien connues et largement utilisées dans les modèles de lisibilité. Sans surprise, elles se sont révélées de bons prédicteurs, en particulier la fréquence, qui suffit pour classer correctement 61% des mots de Manulex et 39% de ceux de FLELex. Les variables morphologiques, l'une des principales voies d'exploration de cette article, ne présentent par contre qu'une faible relation avec la difficulté lexicale, ce qui pourrait être dû à leur acquisition de manière non supervisée. Enfin, nous avons testé deux variables sémantiques qui permettent d'identifier les mots polysémiques et qui présentent des corrélations significatives avec la difficulté du lexique.

Globalement, l'intégration de variables plus complexes que la simple fréquence n'apporte qu'une faible amélioration à notre modèle de prédiction de la complexité lexicale qui procède par apprentissage supervisé. Différentes explications

peuvent être avancées pour rendre compte de ce résultat : d'une part, les données d'apprentissage ne sont pas équilibrées, certaines classes étant bien moins représentées que d'autres dans nos données. Cependant, comme nous l'avons montré, le fait de rééquilibrer le jeu de données par ré-échantillonnage n'améliore pas les performances du modèle. Dès lors, ce qui pénalise le modèle et le pousse à se concentrer sur les classes les plus peuplées semble plutôt être son incapacité à discriminer entre les classes à l'aide des informations qui lui ont été fournies. Il faut donc postuler deux explications à cette situation : soit nos variables ne suffisent pas à expliquer la difficulté des mots, soit les données d'entraînement sont trop bruitées. Très probablement, ces deux phénomènes se combinent. D'une part, on observe que très peu de corrélations fort élevées au sein de notre ensemble de variables. D'autre part, la transformation des données d'origine vers un lexique classé selon des niveaux indépendants nécessite de faire des choix qui ne sont pas sans répercussion sur le modèle de classification. Sans compter que l'apparition d'un mot à un niveau précis d'un manuel reste dépendant des choix des éditeurs et de leur propre vision pédagogique.

Afin de poursuivre l'analyse et la prédiction de la complexité lexicale, nous envisageons plusieurs pistes. Tout d'abord, d'autres variables identifiées dans la littérature psycholinguistique pourraient encore faire l'objet d'expérimentations. Citons parmi celles-ci le caractère abstrait ou concret d'un mot, son voisinage phonologique, ou encore le niveau d'imageabilité. Ensuite, nous pourrions reproduire notre étude sur un lexique plus restreint, mais dont la difficulté aurait été contrôlée plus strictement, via des tests sur populations. L'objectif serait de déterminer dans quelle mesure le plafonnement des performances provient d'un manque informationnelle au sein du modèle ou du bruit dans le corpus. Enfin, une autre approche de la complexité lexicale est possible dans le cadre d'applications de simplification pour lesquelles il faut choisir le meilleur substitut. Il s'agirait de développer un modèle qui ne viserait plus à attribuer un niveau absolu à l'ensemble des mots d'un lexique, mais plutôt de comparer un ensemble restreint de mots sémantiquement liés (par exemple au sein d'un synset) afin de les trier en fonction de leur difficulté.

Références

BERNHARD D. (2006). Unsupervised morphological segmentation based on segment predictability and word segments alignment. In *Proceedings of the PASCAL Challenges Workshop on Unsupervised Segmentation of Words into Morphemes*, p. 19–23.

BERNHARD D. (2010). Apprentissage non supervisé de familles morphologiques : comparaison de méthodes et aspects multilingues. *Traitement Automatique des Langues*, **2**(51), pp. 11–39.

BIRAN O., BRODY S. & ELHADAD N. (2011). Putting it simply : a context-aware approach to lexical simplification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies*, p. 496–501.

BLACHE P. (2011). A computational model for language complexity. In *1st Conference on Linguistics, Biology and Computational Science*, Tarragona, Spain.

BOSER B., GUYON I. & VAPNIK V. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, p. 144–152.

BRYSBAERT M., LANGE M. & VAN WIJNENDAELE I. (2000). The effects of age-of-acquisition and frequency-of-occurrence in visual word recognition : Further evidence from the Dutch language. *European Journal of Cognitive Psychology*, **12**(1), 65–85.

CARROLL J., MINNEN G., CANNING Y., DEVLIN S. & TAIT J. (1998). Practical simplification of English newspaper text to assist aphasic readers. In *Proceedings of the AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*.

CHANDRASEKAR R., DORAN C. & SRINIVAS B. (1996). Motivations and methods for text simplification. In *16th conference on Computational linguistics*, p. 1041–1044.

CHANG C.-C. & LIN C.-J. (2011). Libsvm : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, **2**(3), 27.

COLTHEART M., DAVELAAR E., JONASSON T. & BESNER D. (1977). Access to the internal lexicon. In *Attention and Performance VI*, p. 535–555, London : Academic Press.

CONSEIL DE L'EUROPE (2001). *Cadre européen commun de référence pour les langues : apprendre, enseigner, évaluer*. Paris : Hatier.

DALE E. (1931). A comparison of two word lists. *Educational Research Bulletin*, **18**(10), 484–489.

FAN R.-E., CHANG K.-W., HSIEH C.-J., WANG X.-R. & LIN C.-J. (2008). Liblinear : A library for large linear classification. *The Journal of Machine Learning Research*, **9**, 1871–1874.

FERRAND L. (2007). *Psychologie cognitive de la lecture*. Bruxelles : De Boeck.

FRANÇOIS T. (2011). *Les apports du traitement automatique du langage à la lisibilité du français langue étrangère*. PhD thesis, Université Catholique de Louvain. Thesis Supervisors : Cédrick Fairon and Anne Catherine Simon.

FRANÇOIS T. & FAIRON C. (2012). An "AI readability" formula for French as a foreign language. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing (EMNLP 2012)*, p. 466–477.

FRANÇOIS T., GALA N., WATRIN P. & FAIRON C. (2014). FLELex : a graded lexical resource for French foreign learners. In *Proceedings of International conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Islande.

GALA N., FRANÇOIS T. & FAIRON C. (2013). Towards a French lexicon with difficulty measures : NLP helping to bridge the gap between traditional dictionaries and specialized lexicons. In *E-lexicography in the 21st century : thinking outside the paper.*, Tallin, Estonia.

GALA N. & REY V. (2008). Polymots : une base de données de constructions dérivationnelles en français à partir de radicaux phonologiques. In *TALN 2008, Conférence sur le Traitement Automatique des Langues Naturelles*, Avignon, France.

GALE W. & SAMPSON G. (1995). Good-Turing frequency estimation without tears. *Journal of Quantitative Linguistics*, **2**(3), 217–237.

GERNSBACHER M. (1984). Resolving 20 years of inconsistent interactions between lexical familiarity and orthography, concreteness, and polysemy. *Journal of Experimental Psychology : General*, **113**(2), 256–281.

GOUGENHEIM G. (1958). *Dictionnaire fondamental de la langue française*. Paris : Didier.

HOWES D. & SOLOMON R. (1951). Visual duration threshold as a function of word probability. *Journal of Experimental Psychology*, **41**(40), 1–4.

LAFOURCADE M. (2007). Making people play for Lexical Acquisition. In *Proc. SNLP 2007, 7th Symposium on Natural Language Processing.*, Pattaya, Thailande.

LAUFER B. (1997). *What's in a word that makes it hard or easy : Some intralexical factors that affect the learning of words.* Cambridge University Press.

LÉTÉ B., SPRENGER-CHAROLLES L. & COLÉ P. (2004). Manulex : A grade-level lexical database from French elementary-school readers. *Behavior Research Methods, Instruments and Computers*, **36**, 156–166.

MONSELL S. (1991). The nature and locus of word frequency effects in reading. In D. BESNER & G. HUMPHREYS, Eds., *Basic processes in reading : Visual word recognition*, p. 148–197. Hillsdale, NJ : Lawrence Erlbaum Associates Inc.

NAVIGLI R. & PONZETTO S. P. (2010). BabelNet : building a very large multilingual semantic network. In *48th annual meeting of the Association for Computational Llinguistics.*, p. 216–225, Uppsala, Suède.

NEW G. A., PALLIER C., FERRAND L. & MATOS R. (2001). Une base de données lexicales du français contemporain sur Internet : Lexique 3. *L'année psychologique*, **101**, 447–462.

PALLIER C. (1999). *Syllabation des représentations phonétiques de Brulex et de Lexique*. Rapport interne, Technical Report, update 2004. Lien : http ://www. pallier. org/ressources/syllabif/syllabation. pdf.

PINTO S., GHIO A., TESTON B. & VIALLET F. (2010). La dysarthrie au cours de la maladie de parkinson. histoire naturelle de ses composantes : dysphonie, dysprosodie et dysarthrie. *Revue Neurologique*, **166**(10), 800–810.

QUASTHOFF U., RICHTER M. & BIEMANN C. (2006). Corpus portal for search in monolingual corpora. In *Proceedings of the fifth international conference on Language Resources and Evaluation, LREC 2006*, p. 1799–1802, Genoa, Italy.

ROMARY L., SALMON-ALT S. & FRANCOPOULO G. (2004). Standards going concrete : from LMF to Morphalou. In *Workshop on Electronic Dictionaries, COLING, Conference on Computational Linguistics*, Geneva, Suisse.

SCHREUDER R. & BAAYEN H. (1997). How complex simplex words can be. *Journal of Memory and Language*, p. 118–139.

SPECIA L., JAUHAR S. K. & MIHALCEA R. (2012). Semeval-2012 task 1 : English lexical simplification. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), Montreal, Canada*.

THORNDIKE E. (1921). *The Teacher's Word Book*. New York : Teachers College.

VITU F., O'REGAN J. & MITTAU M. (1990). Optimal landing position in reading isolated words and continuous text. *Perception & Psychophysics*, **47**(6), 583–600.

Annotations et inférences de relations dans un réseau lexico-sémantique: application à la radiologie

L Ramadier[1, 2] M Zarrouk[1] M Lafourcade[1] A Micheau[2]
(1) LIRMM, 161, rue ADA 34392 Montpellier Cedex 5
(2) IMAIOS, 34090 Montpellier
lionel.ramadier@lirmm.fr,manel.zarrouk@lirmm.fr, mathieu.lafourcade@lirmm.fr
antoine.micheau@imaios.com

Résumé. Les ontologies spécifiques à un domaine ont une valeur inestimable malgré les nombreux défis liés à leur développement. Dans la plupart des cas, les bases de connaissances spécifiques à un domaine sont construites avec une portée limitée. En effet, elles ne prennent pas en compte les avantages qu'il pourrait y avoir à combiner une ontologie de spécialité à une ontologie générale. En outre, la plupart des ressources existantes manque de méta-informations sur les annotations (informations fréquentielles : de fréquent à rare ; ou des informations de pertinence : pertinent, non pertinent et inférable). Nous présentons dans cet article un réseau lexical dédié à la radiologie construit sur un réseau lexical généraliste (JeuxDeMots). Ce réseau combine poids et annotations sur des relations typées entre des termes et des concepts, un mécanisme d'inférence et de réconciliation dans le but d'améliorer la qualité et la couverture du réseau. Nous étendons ce mécanisme afin de prendre en compte non seulement les relations mais aussi les annotations. Nous décrivons la manière de laquelle les annotations améliorent le réseau en imposant de nouvelles contraintes spécialement celles basées sur la connaissance médicale. Nous présentons par la suite des résultats préliminaires.

Abstract. Relations annotation and inference in a lexical-semantic network : application to radiology Domain specific ontologies are invaluable despite many challenges associated with their development. In most cases, domain knowledge bases are built with very limited scope without considering the benefits of plunging domain knowledge to a general ontology. Furthermore, most existing resources lack meta-information about association strength (weights) and annotations (frequency information like frequent, rare ... or relevance information (pertinent or irrelevant)). In this paper, we are presenting a semantic resource for radiology built over an existing general semantic lexical network (JeuxDeMots). This network combines weight and annotations on typed relations between terms and concepts. Some inference mechanisms are applied to the network to improve its quality and coverage. We extend this mechanism to relation annotation. We describe how annotations are handled and how they improve the network by imposing new constraints especially those founded on medical knowledge. We present then some results

Mots-clés : réseau lexical, inférence, annotation, radiologie.

Keywords: relation inference, lexical semantic network, relation annotation, radiology.

1 Introduction

Depuis environ deux décennies, l'utilisation d'ontologies et de réseaux lexicaux dans le domaine biomédical est devenue très répandue (Bodenreider *et al.*, 2008). Ces ressources sont utilisées pour l'analyse sémantique comme pour la reconnaissance d'entités nommées (par exemple l'identification de noms des gènes) ou bien l'extraction de relations (identification des relations sémantiques entre entités biomédicales (Abacha et Zweigenbaum, 2011) comme pour le cas des interactions entre protéines). Dans le cadre du projet UMLS (Unified Medical Language System), un réseau sémantique a été construit (Lomax et McCray, 2004). Ce réseau est utilisé dans le domaine de la radiologie pour analyser de façon automatique les comptes rendus radiologiques afin d'extraire les recommandations en vue d'améliorer la prise en charge des patients (Yetisgen-Yildiz *et al.*, 2013). La plupart du temps, l'ontologie dédiée à la radiologie est plongée dans une ontologie médicale généraliste qui est trop importante et complexe pour l'utilisateur final. Pour tenter de résoudre ce problème, la société de radiologie Nord-Américaine (RSNA) a crée une ontologie spécifiquement dédiée à la radiologie Radlex (Rubin, 2008), (Mejino Jr *et al.*, 2008). Cependant, la couverture de RadLex n'est pas considérée comme complète (Hong *et al.*, 2012). Il existe une version allemande de RadLex (Gerstmair *et al.*, 2012) mais aucune en français à notre connaissance.

Cependant, un lexique médical unifié en langue française a été réalisé (Zweigenbaum *et al.*, 2005) mais il reste d'ordre général.

Dans le domaine de la radiologie où il peut être intéressant d'extraire des termes pertinents des comptes-rendus et les relier aux images (Napel *et al.*, 2010), les relations pertinentes entre les termes sont cruciales et les modèles taxonomiques ne capturent pas ces informations aussi bien qu'un réseau sémantique, la taxonomie indiquant seulement la hiérarchie entre les termes (relation is-a). Il peut être intéressant pour le médecin de disposer également plus facilement de relations non hiérarchiques. Par exemple, il est pertinent de donner pour une certaine maladie la liste des symptômes, des cibles potentielles, des localisations anatomiques et cela indépendamment de toute hiérarchie. Ceci peut être modélisé de façon plus simple par un réseau sémantique. L'association entre un réseau sémantique général et spécialisé peut jouer un rôle important dans l'analyse des comptes rendus radiologiques. En effet, dans la section *Indication* du rapport de radiologie, le texte est souvent écrit avec des termes courants alors que la section *résultats* comporte des termes très spécialisés. Le but de la construction d'un tel réseau est d'analyser les comptes rendus radiologiques dans leur totalité et d'en extraire les termes importants mais également les relations sémantiques pertinentes. Cette extraction pourra servir à annoter et indexer le texte et indirectement les images médicales afin de faciliter leur recherche et utilisation.

La construction d'un réseau lexico-sémantique peut être réalisée soit manuellement soit via une analyse de corpus. Par exemple, ConceptNet qui est une base de connaissance générale, est générée automatiquement à partir de 700 000 phrases du Open Mind Common Sense Project (Liu et Singh, 2004). Mais les approches entièrement automatisées sont généralement limitées à la co-occurence des termes car l'extraction des relations sémantiques précises entre termes à partir d'un texte reste difficile. Dans l'optique de la création d'un réseau spécialisé, nous avons décidé d'utiliser JeuxDeMots (JDM) (Lafourcade, 2007) comme base pour le réseau de connaissance générale. Le réseau JeuxDeMots est un réseau lexical construit à partir d'un ensemble de jeux en ligne. Pour la construction du réseau spécialisé, nous avons utilisé Diko un outil contributif proposé par la plateforme JeuxDeMots. La nécessité de ne pas dépendre uniquement de jeux pour construire un réseau lexical dédié à la radiologie vient du fait qu'une partie non négligeable des types de relations de JDM soit sont difficiles à saisir pour un joueur non expert, soit sont peu lexicalisées. Diko utilise par ailleurs des mécanismes d'inférences (Zarrouk *et al.*, 2013b) pour proposer automatiquement de nouvelles relations à partir de celles qui existaient déjà dans le réseau. Cette approche est strictement endogène et ne prend pas en compte des ressources externes.

JDM se fonde sur la peuplonomie (crowdsourcing) pour établir les poids des relations entre les termes. Chaque occurrence de relation est pondérée indiquant la force d'association (elle représente le nombre de joueurs qui ont pensé pour une relation donnée au même terme, à la même position parmi la liste des mots qu'ils ont proposés). Pour certains concepts ou termes, certaines idées viendront spontanément à l'esprit de beaucoup d'utilisateurs. La force d'association sera alors importante. Cette approche consistant à attribuer des poids à une relation est bien adaptée pour les connaissances générales et l'association de termes. Notons qu'il n'y a pas systématiquement une corrélation entre l'importance de la relation pour un domaine considéré et sa force d'association. Pour remédier à ce problème, nous introduisons des annotations entre certaines relations dans le réseau lexico-sémantique. Le but de ces annotations est de guider et d'améliorer le processus d'inférence et d'analyse sémantique.

Dans cet article, nous présentons les principes de construction du réseau lexical et nous l'illustrons grâce au projet JeuxDeMots. Nous discutons aussi de la construction d'un réseau spécialisé en imagerie médicale ainsi qu'un mécanisme d'inférence à savoir le schéma déductif. Ensuite nous détaillons le principe des annotations des relations entre termes médicaux. Dans une dernière partie nous décrivons nos expériences et les premiers résultats obtenus. Nous concluons avec les perspectives et les pistes futures pour la recherche.

2 Réseaux lexicaux et inférences

Un réseau lexico-sémantique est un graphe orienté, pondéré, typé avec des sommets qui représentent les concepts et des arcs les relations entre ces concepts. Il existe plusieurs méthodes pour construire un réseau lexical en tenant compte des facteurs principaux tels que la qualité des données, le coût et le temps de développement. Les approches contributives connaissent une forte popularité car elle se révèlent à la fois peu coûteuses et efficaces en qualité. L'intérêt porté au GWAP (games with purpose ou human-based computation game) comme méthode d'acquisition de ressources variées augmente régulièrement (Thaler *et al.*, 2011).

Le réseau JDM est un réseau lexico-sémantique construit à partir d'un ensemble de jeux en ligne. 6 790 189 relations et 316 983 termes sont présents dans la base. Pour le terme *médecine*, il existe 10 112 relations dans le réseau lexical. Environ 350 relations ont été créées pour le mot *IRM* (figure1).

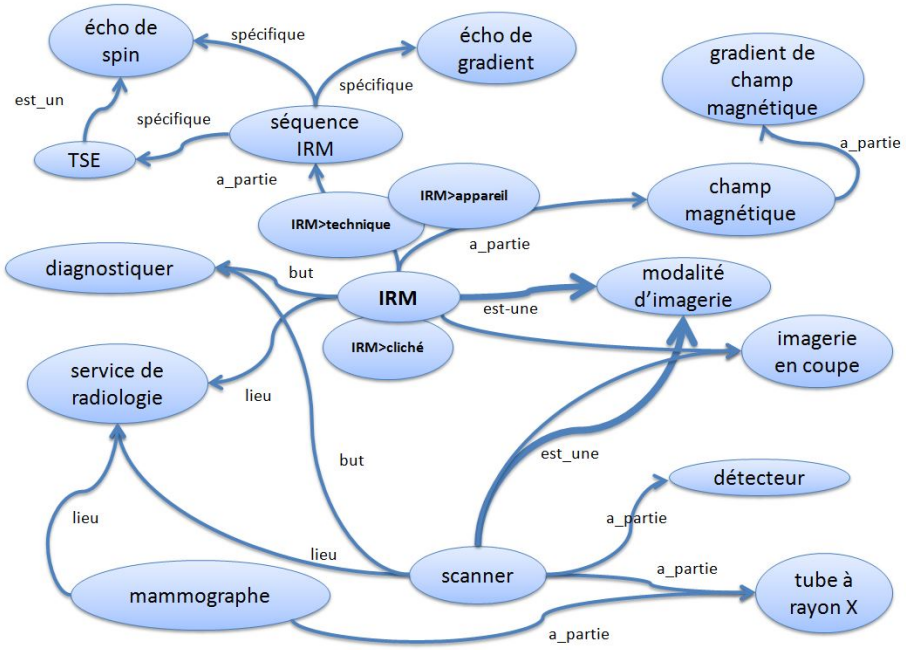


FIGURE 1 – Exemple de sous réseau lexical dédié à la radiologie/médecine : une partie des relations qu'entretient le terme *IRM*.

2.1 Le modèle JeuxDeMots

JeuxDeMots est un GWAP (Lafourcade, 2007) associant les joueurs par paires, et visant à construire un grand réseau lexico-sémantique. Il y a plus de 50 types de relations et chaque occurence de relation est pondérée indiquant une force d'association. Au début d'une partie, une consigne concernant le type de la relation (*idée associé*, *générique*, *caractéristique*, etc) ainsi qu'un terme cible issu du réseau lexical (par exemple donner des *idées associées* aux termes *maladie*) sont présentés au joueur. Ce dernier a un temps limité pour saisir des termes qui lui semblent correspondre à la consigne. Par la suite, ce même couple terme/consigne est proposé à d'autres joueurs. Les réponses communes par paires de joueurs sont insérées dans le réseau lexical ou le renforcent, c'est à dire dans le cas où la relation entre deux termes existe déjà, son poids est augmenté (force d'association). Le jeu est très efficace pour les connaissances générales mais l'est un peu moins pour les connaissances spécialisées. En effet, la majorité des joueurs n'ont pas de connaissance particulière dans le domaine de la radiologie. c'est pourquoi nous utilisons un autre outil fourni par JDM : Diko.

2.2 Diko : un outil contributif pour JDM

Diko est un outil en ligne sur le web permettant de visualiser les informations contenues dans le réseau lexical JDM, mais également constitue un outil contributif et de vérification. Dans le cadre de la construction d'un réseau dédié à un domaine spécifique, nous utilisons Diko comme un outil de développement d'une base de connaissance pour ce domaine. Le principe du processus de la contribution est qu'une proposition faite par un expert en radiologie sera soumise aux votes d'autres experts en imagerie médicale ou en médecine pour un processus de validation/invalidation. Dans le champ de la médecine, nous avons ajouté certains types de relations comme par exemple : *symptômes* ou *diagnostic*. Il nous paraît intéressant dans une base de connaissances dédiées à la radiologie de préciser pour une maladie donnée ses *symptômes* (cliniques), la population cible (*cible*), de même que les moyens de diagnostic (*diagnostic*). Cela peut avoir un intérêt pour l'expansion de requêtes dans l'analyse ou la recherche d'information (recherche d'image pour les patients présentant les symptômes de telle ou telle maladie). La construction d'un réseau spécialisé dans le domaine de l'imagerie médicale a été réalisée à partir d'un corpus de 40 000 compte rendus radiologiques représentant les différentes modalités d'imagerie médicale (imagerie par résonance magnétique, tomodensitométrie à rayon X, artériographie, échographie). La première étape a consisté à réaliser un index inversé de bigrame, trigrame à partir du corpus. Dans un deuxième temps, l'expert a soumis aux autres spécialistes du domaine les termes qu'il a jugés pertinents pour un processus de validation/invalidation. La majorité des concepts génériques a pu être rattachés aux concepts radiologiques(la relation *lieu*, *has part* etc..). Ce

travail contributif est nécessaire pour construire une base de connaissance liée à la radiologie.

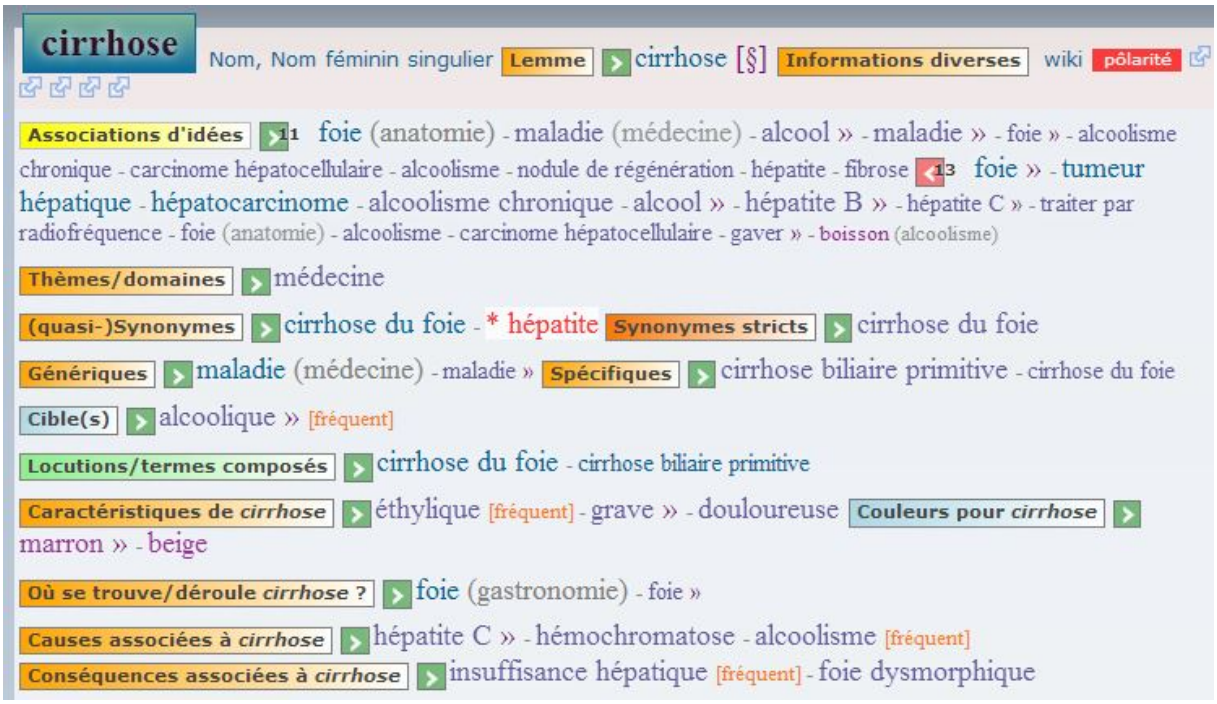


FIGURE 2 – Capture écran de la fenêtre de Diko du terme *cirrhose*, par exemple, *cause alcoolisme* annotée comme *fréquent*.

Dans le but d'améliorer la pertinence des informations sémantiques du réseau, nous ajoutons des annotations à certaines relations entre termes, en particulier pour ce qui nous concerne dans ce travail ceux liés à la médecine. Par exemple, pour la relation suivante *cirrhose* (cause) *alcoolisme* nous ajoutons l'annotation *fréquent* (figure 2). Nous donnons un autre exemple pour le terme *sclérose en plaques* (figure 3). Dans la troisième partie, nous détaillons le concept d'annotations de relations.

Dans le but de formuler de nouvelles conclusions (c'est à dire des relations entre les termes) à partir de prémisses (des relations préexistantes), un moteur d'inférence a été proposé (Zarrouk *et al.*, 2013b). Le moteur d'inférence propose des relations, à l'image d'un contributeur, qui vont être votées par la suite par un autre contributeur et validées par un expert dans le domaine de l'imagerie médicale. Dans le cadre de ce travail nous décrivons un seul type d'inférence : le schéma déductif. Le schéma déductif est basé sur la transitivité de la relation ontologique *is-a* (hyperonyme). Si un terme A est un type de B et B a une relation R avec le terme C, alors on peut proposer que A entretienne la même relation avec C. Le moteur d'inférence est appliqué sur les termes ayant au minimum un hyperonyme. Si un terme T possède un ensemble d'hyperonymes pondérés, le moteur d'inférence déduit un ensemble d'inférences. Ces hyperonymes vont être classés selon un ordre hiérarchique. Le poids d'une inférence proposée est la moyenne géométrique incrémentale de chaque occurrence (c'est à dire que la présence d'un poids négatif suffit à rendre la moyenne invalide). Le schéma présenté ci dessus est très simple, en effet le terme B peut être polysémique, et l'inférence proposée sera probablement fausse. Nous utilisons alors un blocage logique (figure 4). Ce mécanisme a été décrit dans un précédent travail (Zarrouk *et al.*, 2013b)

Dans le cas d'invalidation, un agent réconciliateur est invoqué pour essayer d'évaluer pourquoi la relation a été trouvé fausse : erreur dans les prémisses, polysémie (l'inférence est faite en se basant sur un terme central polysémique) ou une exception. Dans ce qui suit, c'est ce type d'inférence que nous allons considérer. Néanmoins, il existe deux autres types d'inférences : l'induction (du spécifique au général) et l'abduction (imitation par des exemples similaires)... .

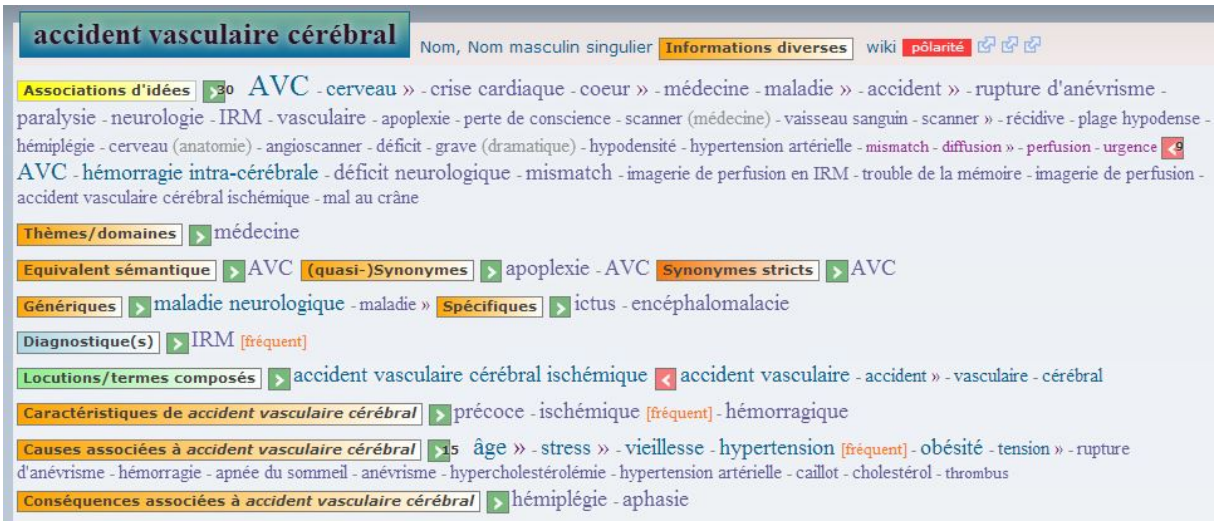


FIGURE 3 – Fenêtre de Diko du terme *accident vasculaire cérébral* dont la *caractéristique ischémique* est *fréquente*.

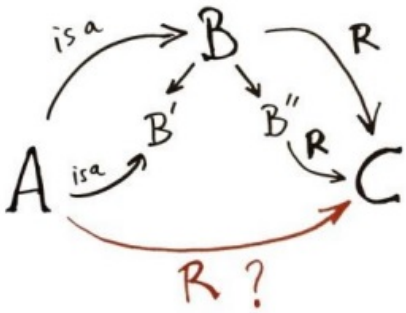


FIGURE 4 – Schéma d'inférence déductive triangulaire avec un blocage logique se basant sur la polysémie du terme B du milieu. Les termes B' et B" sont des raffinements/usages de B.

.

3 Annotation de relations

En général, et surtout dans le domaine des connaissances spécialisées, la corrélation entre la force d'association de la relation et son importance conceptuelle n'est pas toujours assurée. Par exemple, pour le terme *carcinome hépatocellulaire*, la relation *caractéristique* avec *wash-out* est très spécifique à la radiologie, par conséquence le poids de la relation sera faible dans le cadre général de la médecine mais pour le radiologue cette relation est particulièrement importante. Un autre cas est la relation *diagnostic* entre la *sclérose en plaques* et l'*IRM*. Nous avons affaire à une relation, là encore, spécifique au domaine de l'imagerie médicale, qui sera pertinente pour le radiologue. C'est pourquoi, il est apparu intéressant d'introduire des annotations pour certaines relations. Dans le réseau lexical, une relation est représentée par un triplet : $< noeud_{source}$, type de relation / annotation, $noeud_{cible} >$
Dans le champ de la radiologie, les relations les plus utiles pour le radiologue, qui ont été établies par des radiologues suivant leur pratique quotidienne, sont indiquées dans le tableau 1.Parmi les relations pertinentes, seules trois ont été rajoutées(*symptômes*, *diagnostique* et *cibles*), toutes les autres proviennent du domaine général. Dans les ontologies existantes dédiées à la radiologie comme RadLex, il n'existe pas autant de type de relations potentiellement utiles pour l'analyse des comptes rendus. Dans la recherche d'informations radiologiques (comptes-rendus et images), ces annotations peuvent apporter un complément d'information et permettre de classer les réponses par ordre de pertinence. Par exemple cela peut aider les radiologues devant une image anormale pour savoir si une caractéristique est *rare* ou *fréquente* et ainsi leur apporter une aide au diagnostic. D'autres types d'annotations peuvent exister comme par exemple la pertinence ou non d'une relation entre deux termes. Mais ce type d'information est généralement absent d'un réseau ou d'une ontologie. Par exemple, la relation *caractéristique* entre *carcinome hépatocellulaire* et *hypervasculaire* est *fréquent* et cette information

is-a	Hyperonymes du terme. Exemple : *IRM* est une *modalité d'imagerie* (possible)
partie-de	Parties, constituants, éléments du mot cible. Exemple : *foie* a comme partie *segment I* (toujours vrai)
caractéristique	Caractéristiques (adjectifs) possibles,typiques. Exemple : *carcinome hépatocellulaire* carac *hypervasculaire* (fréquent)
localisation	Lieux typiques où peut se trouver le terme/objet en question. Exemple : *sclérose en plaque* loc *système nerveux central*
cible	Population affecté par le terme. Exemple : *rougeole* cible *enfant* (fréquent)
diagnostic	Examen. Exemple : *sclérose en plaque* diag *IRM* (fréquent, crucial)
symptôme	Symptômes d'une maladie, *rougeole* symptôme *fièvre* (fréquent)
cause	B est une cause de A. Exemple : *cirrhose* cause *alcoolisme*
conséquence	B est une conséquence possible de A. Exemple : *accident vasculaire cérébral* peut avoir comme conséquence une *hémiplégie*

TABLE 1 – Relations pertinentes en radiologie pour l'analyse de compte-rendu

sera directement disponible dans le réseau (figure 5). Ces annotations ont par ailleurs, une fonction de filtre dans le schéma d'inférence.

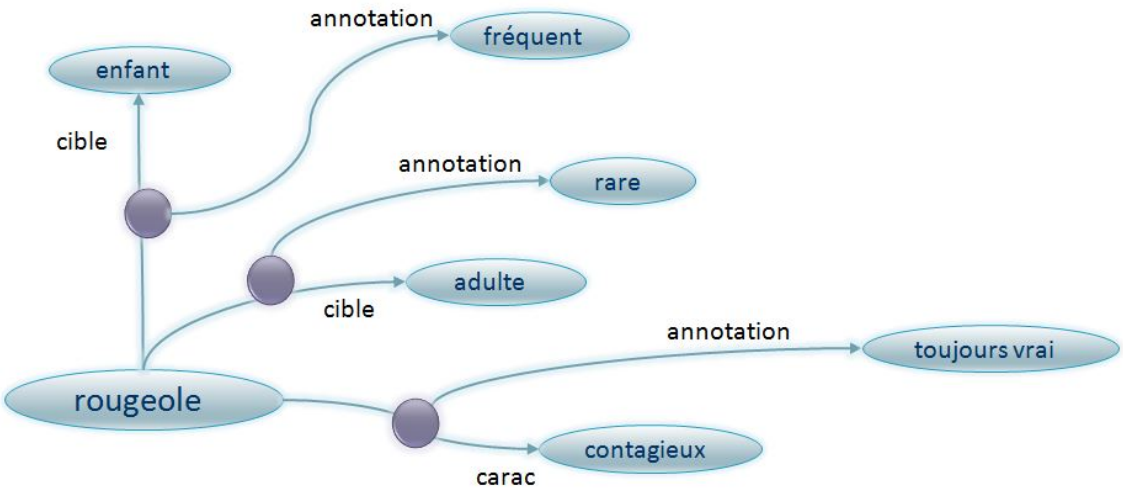


FIGURE 5 – Exemple d'implémentation d'annotation. L'implémentation d'une annotation se fait par réification de la relation à annoter dans le réseau lexical. Le noeud relation ainsi créé peut être associé à d'autres termes. La relation annotation n'est qu'un type de relation parmi d'autres. Les valeurs d'annotation sont des termes standard.

.

Les types d'annotations peuvent être de nature différente (information fréquentielle d'usage ou de pertinence). Ci-dessous, nous présentons les principaux types d'annotation :

- annotations fréquentielles : *très rare*, *rare*, *possible*, *fréquent*, *toujours vrai* ;
- annotations d'usage : *souvent crû vrai*, *abus de langage* ;
- annotations quantitatives : *un nombre (1, 2, 4, ...)*, *beaucoup*, *peu*, etc ;
- annotations d'exception : *exception* ;
- annotations qualitatives : *pertinent*, *non pertinent*, *inférable*.

Un médecin peut utiliser le terme *grippe* au lieu de *virus de la grippe* : c'est un **abus de langage**, le praticien fait simplement un raccourci de langage sans pour autant faire de confusion dans son esprit. Il semble évident pour lui que ces deux expressions sont différentes. L'annotation **souvent crû vrai** s'applique pour une fausse relation (avec un poids négatif) qui est souvent considérée comme vrai, par exemple *araignée* (*is-a/souvent crû vrai*) *insecte*. Les exceptions sont également renseignées et prennent la forme d'une relation ayant un poid négatif. Ce type d'annotation est utilisé pour

bloquer le schéma d'inférence.

L'annotation de nature qualitative est liée au statut inférable de la relation, particulièrement concernant l'inférence. L'annotation **pertinente** se rapporte à un niveau ontologique adéquat pour une relation donnée. Par exemple, *être vivant (carac/pertinent) vivant* ou *être vivant (carac/pertinent) mort*. L'annotation **inférable** est supposée être ajoutée quand une relation est inférable (ou a été inférée) à partir d'une relation existante, par exemple : *chien (carac/inférable) vivant* car *chien (is-a) être vivant*. L'annotation *non pertinent* est ajoutée aux relations vraies mais qui sont très éloignées du niveau pertinent, par exemple *animal (possède/non pertinent) atomes*. Pour avoir l'annotation la plus précise, nous avons besoin d'ordonner les termes centraux du plus spécifique au moins spécifique. Pour le terme *carcinome hépatocellulaire*, la hiérarchie sera :

carcinome hépatocellulaire
< tumeur maligne du foie < tumeur du foie < pathologie hépatique < pathologie

sclérose en plaques
< maladie du système nerveux central < neuropathie < maladie dégénérative < maladie

Pour choisir la bonne annotation de la nouvelle relation inférée, la hiérarchie ontologique joue un rôle important. L'annotation du terme le plus spécifique doit avoir plus d'influence que le moins spécifique. Nous prenons en compte ce fait pour les mécanismes d'inférences avec annotations.

Dans le mécanisme d'inférence, le terme B (le terme central) joue un rôle primordial. Nous inspectons la hiérarchie des termes B selon laquelle une relation spécifique a été inférée plusieurs fois et nous gardons la plus spécifique. Si nous obtenons deux termes ou plus ayant le même niveau sémantique, nous appliquons la règle du maximum aux valeurs correspondant à chaque annotation (toujours vrai : 5, fréquent : 4, possible : 3,rare : 2, très rare :1 ...) (figure 6).

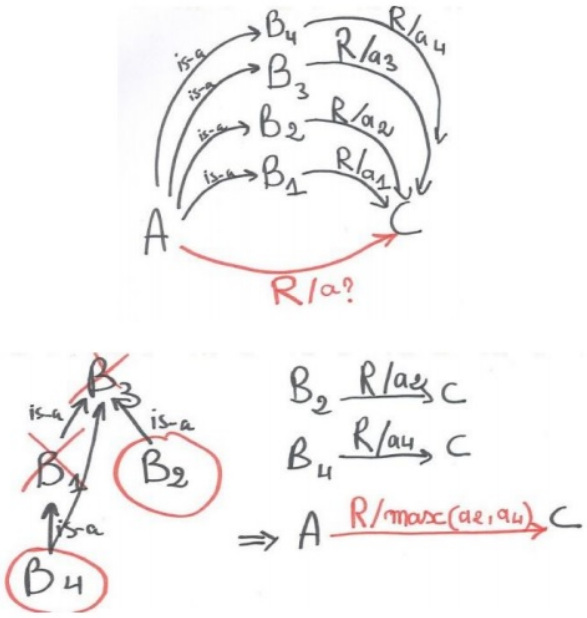


FIGURE 6 – Approche basée sur la hiérarchie utilisée pour choisir l'annotation la plus précise avec plusieurs termes centraux.

4 Expérimentation

Dans une précédente expérience menée par (Zarrouk *et al.*, 2013a), le moteur d'inférence déductive a été appliqué à l'ensemble du réseau lexical. Dans notre approche, nous avons lancé l'expérience sur une partie du réseau lexical JDM qui contient toutes les relations *is-a* (sur laquelle est fondé le schéma déductif) et toutes les relations annotées manuellement et ce dans le but de réduire l'espace de recherche.

4.1 Inférence des relations

Pour augmenter la précision des résultats et pour éviter d'inférer certaines relations peu pertinentes mais vraies (*homme a pour partie protons*), nous avons bloqué les inférences sur les relations qui avaient été annotées comme *non pertinent* ou *exceptionnel*. Le moteur d'inférence déductive a été appliqué sur **146 934** relations produisant un total de **1 825 933** relations avec **573 613** distinctes ce qui fait une moyenne de 3 occurrences par relation (table 2). Il est intéressant de constater que l'inférence renforce le niveau de confiance d'une relation déjà existante.

relations existantes	146934
relations inférées	1825933
relations inférées distinctes	573613

TABLE 2 – Nombre de relations inférées à partir de celles déjà existantes.

4.2 Propagation de l'annotation des relations

Le moteur d'inférence d'annotations est appliqué dans la seconde partie de notre système. Il permet d'ajouter des annotations aux relations de façon automatique à partir d'annotations de relations déjà existantes. Il est lancé sur la base des relations déjà enrichies avec le mécanisme d'inférence déductive. Contrairement au moteur d'inférence, nous autorisons la redondance en vue d'améliorer la précision des résultats du système de propagation d'annotation de relations. Prenons un exemple :

Prémisses : *accident vasculaire cérébral(is-a)infarctus cérébral*
& infarctus cérébral(diagnostic/fréquent) IRM
→ **relation inférée** : *accident vasculaire cérébral (diagnostic/possible) IRM* (1)

Prémisses : *accident vasculaire cérébral(is-a) maladie cérébrovasculaire*
& maladie cérébrovasculaire(diagnostic/possible) IRM
→ **relation inférée** : *accident vasculaire cérébral(diagnostic/possible) IRM* (2)

Le système d'annotations produit deux occurrences **(1)** et **(2)** de la même relation *accident vasculaire cérébral (diagnostic) IRM*, avec deux annotations différentes (possible, fréquent), nous décidons de garder celui avec la plus forte valeur (fréquent). Le système d'inférence des annotations appliqué sur la base de relations provenant des résultats du moteur d'inférence déductive, a annoté **10 085** relations à partir d'une amorce de seulement **72** relations annotées (table 3).

Type d'annotation	**annotation existante**	**annotation inférée**
Fréquentiel : toujours vrai	20	8092
Fréquentiel : fréquent	18	1
Fréquentiel : possible	16	150
Fréquentiel : rare et très rare	7	35
Qualitatif : souvent crû vrai	1	7
Qualitatif : non pertinent	5	1604
Quantificateur :	5	178
Total	72	10085

TABLE 3 – Nombre d'annotations inférées après application du système d'annotation des relations sur celles existantes.

Nous nous concentrons essentiellement sur les annotations concernant la fréquence car ces dernières comportent des informations importantes dans le domaine de la radiologie. Le nombre de relations annotées par type d'annotation ne dépend pas du nombre de relations existantes au départ mais simplement du nombre de relations d'hyperonymie existantes pour le terme central. Le schéma d'inférence est le suivant :

A(is-a)**B** et **B**(R/annot)**C**→**A**(R/annot)**C**

Par exemple :

$$\underbrace{\begin{array}{c}\text{cancer du poumon non à petites cellules}\\ \text{carcinome hépatocellulaire}\\ \text{glioblastome}\end{array}}_{\begin{array}{c}\text{(is-a) tumeur maligne}\\ \text{\& tumeur maligne (carac/frequent) mauvais pronostic}\end{array}} \quad (1)$$

Plus le nombre de relations d'hyperonymie vers le terme B (*tumeur maligne*) qui a une relation annotée (tumeur maligne(carac/fréquent)) est important, plus le nombre de relations annotées est élevé. Supposons que le terme *carcinome hépatocellulaire* n'ai pas de relation d'hyperonymie, donc dans ce cas l'annotation *fréquent* ne générera pas d'autre annotation. Ceci peut expliquer la raison pour laquelle il y a peu d'annotations inférées pour le type d'annotation fréquentiel *fréquent*. Notons que l'absence de certaines relations ou certains termes est due à l'aspect de progression continue du réseau qui fait qu'il est possible qu'à un instant précis un terme ou une relation manquent.

Nous avons évalué le nombre d'annotations inférées, et il apparaît que 87% d'entre elles ont été évaluées "correctes", 5% comme "incorrectes" et le reste (8%) comme "discutable" (les experts discuteront non pas leur validité mais plutôt leur valeurs fréquentielles pour savoir si elles doivent être modifiées). Dans cette expérience, nous avons appliqué le système relation/annotation une seule fois sur l'ensemble du réseau lexical. Évidemment, comme le réseau est en construction permanente, et que le partie consacrée à la radiologie n'en est qu'à ses débuts, de nouveaux termes ainsi que de nouvelles annotations seront rajoutées. Le système d'inférences et d'annotations tournent à présent en continu dans le but de consolider notre réseau lexico-sémantique.

5 Conclusion

Dans cet article, nous avons présenté quelques éléments pour la construction d'une base de connaissance spécialisée (en radiologie) dans un réseau lexical général et en particulier un modèle d'inférence et d'annotation de relations. Pour améliorer la qualité du réseau et sa couverture, nous avons proposé une approche de consolidation basée sur un moteur d'inférence réalisé sur des relations annotées. Le système d'annotation décrit dans cet article peut être vu comme un complément du système de consolidation de réseau lexico-sémantique. Ce système propage, grâce à la procédure d'annotation, des informations sémantiques ou d'usages importants qui peuvent être utilisées non seulement dans le domaine de la radiologie comme illustré dans cet article mais aussi dans d'autres domaines de spécialités. Il nous semble intéressant de développer des bases de connaissances dans des domaines spécialisés plongée dans un réseau lexical de sens commun. De futures recherches doivent également viser à améliorer la diffusion des annotations de relations à travers le réseau mais aussi améliorer le lexique spécialisé en radiologie à l'aide non pas seulement des experts mais aussi de non experts. Ce réseau lexico-sémantique nous sert pour les analyses sémantiques et d'indexation de comptes-rendus radiologiques. Cette analyse sémantique nous permet d'extraire des relations entre des concepts et des termes médicaux. Elle pourra être combinée avec la recherche d'image par le contenu (Content Based Image Retrieval ou CBIR) qui constitue une piste de recherche pour nos prochain travaux. En effet cette dernière technique pourra être combinée avec une recherche sémantique dans le but d'améliorer la recherche d'information dans le domaine de l'imagerie médicale.

Références

ABACHA, A. B. et ZWEIGENBAUM, P. (2011). A hybrid approach for the extraction of semantic relations from medline abstracts. *Computational Linguistics and Intelligent Text Processing*, pages 139–150.

BODENREIDER, O. *et al.* (2008). Biomedical ontologies in action : role in knowledge management, data integration and decision support. *Yearb Med Inform*, 47:67–79.

GERSTMAIR, A., DAUMKE, P., SIMON, K., LANGER, M. et KOTTER, E. (2012). Intelligent image retrieval based on radiology reports. *European radiology*, 22(12):2750–2758.

HONG, Y., ZHANG, J., HEILBRUN, M. E. et KAHN JR, C. E. (2012). Analysis of radlex coverage and term co-occurrence in radiology reporting templates. *Journal of Digital Imaging*, 25(1):56–62.

LAFOURCADE, M. (2007). Making people play for lexical acquisition with the jeuxdemots prototype. *SNLP'07 : 7th International Symposium on Natural Language Processing. Pattaya, Thailande*, page 8p.

LIU, H. et SINGH, P. (2004). Conceptnet - a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4): 211–226.

LOMAX, J. et MCCRAY, A. T. (2004). Mapping the gene ontology into the unified medical language system. *Comparative and functional genomics*, 5(4):354–361.

MEJINO JR, J. L., RUBIN, D. L. et BRINKLEY, J. F. (2008). Fma-radlex : An application ontology of radiological anatomy derived from the foundational model of anatomy reference ontology. *AMIA Annual Symposium Proceedings*, 2008:465.

NAPEL, S. A., BEAULIEU, C. F., RODRIGUEZ, C., CUI, J., XU, J., GUPTA, A., KORENBLUM, D., GREENSPAN, H., MA, Y. et RUBIN, D. L. (2010). Automated retrieval of ct images of liver lesions on the basis of image similarity : method and preliminary results. *Radiology*, 256(1):243.

RUBIN, D. L. (2008). Creating and curating a terminology for radiology : ontology modeling and analysis. *Journal of digital imaging*, 21(4):355–362.

THALER, S., SIORPAES, K., SIMPERL, E. et HOFER, C. (2011). A survey on games for knowledge acquisition. *Rapport technique, STI*, page 26.

YETISGEN-YILDIZ, M., GUNN, M. L., XIA, F. et PAYNE, T. H. (2013). A text processing pipeline to extract recommendations from radiology reports. *Journal of biomedical informatics*, 46(2):354–362.

ZARROUK, M., LAFOURCADE, M. et JOUBERT, A. (2013a). Inductive and deductive inferences in a crowdsourced lexical-semantic network. *9th International Conference on Recent Advances in Natural Language Processing (RANLP 2013)*, page 6p.

ZARROUK, M., LAFOURCADE, M. et JOUBERT, A. (2013b). Inference and reconciliation in a lexical-semantic network. *14th International Conference on Intelligent Text Processing and Computational Linguistic (CICLING-2013)*, page 13p.

ZWEIGENBAUM, P., BAUD, R., BURGUN, A., NAMER, F., JARROUSSE, É., GRABAR, N., RUCH, P., LE DUFF, F., FORGET, J.-F., DOUYERE, M. *et al.* (2005). Umlf : a unified medical lexicon for french. *International Journal of Medical Informatics*, 74(2):119–124.

Correction automatique par résolution d'anaphores pronominales

Maud Pironneau, Éric Brunelle, Simon Charest
Druide informatique, 1435 rue Saint-Alexandre, bureau 1040, Montréal (Québec)
developpement@druide.com

Résumé. Cet article décrit des travaux réalisés dans le cadre du développement du correcteur automatique d'un logiciel commercial d'aide à la rédaction du français. Nous voulons corriger des erreurs uniquement détectables lorsque l'antécédent de certains pronoms est connu. Nous décrivons un algorithme de résolution des anaphores pronominales intra- et interphrastiques s'appuyant peu sur la correspondance de la morphologie, puisque celle-ci est possiblement erronée, mais plutôt sur des informations robustes comme l'analyse syntaxique fine et des cooccurrences fiables. Nous donnons un aperçu de nos résultats sur un vaste corpus de textes réels et, tout en tentant de préciser les critères décisifs, nous montrons que certains types de corrections anaphoriques sont d'une précision respectable.

Abstract. This article relates work done in order to expand the performance of a commercial French grammar checker. We try to achieve the correction of errors only detectable when an anaphora pronoun is linked with its antecedent. The algorithm searches for the antecedent within the same sentence as the pronoun as well as in previous sentences. It relies only slightly on morphology agreement, since it is what we are trying to correct, and uses instead information from a robust syntactic parsing and reliable cooccurrences. We give examples of our results on a vast corpus, and try to identify the key criteria for successful detection. We show that some types of corrections are precise enough to be integrated in a large scale commercial software.

Mots-clés : Correcteur, Anaphores, Pronom, Saillance, Approche multistratégique, Cooccurrences.
Keywords: Grammar checker, Anaphora, Pronoun, Salience, Multi-Strategy Approach, Cooccurrences.

1 Introduction

1.1 Objet des travaux

L'objectif est la correction de certains types d'erreurs, et non pas la reconnaissance générale des entités.

(1a) O : {*Alice*} n'a rien mangé ce *matin*. Je **l**'ai LEVÉ et **l**'ai EMMENÉ à la *garderie*.

La seconde phrase de l'exemple ci-dessus comporte deux erreurs d'accord. On ne peut détecter ces erreurs que lorsque l'on relie les deux pronoms *l'*, avec lesquels s'effectue l'accord des deux participes passés, au nom propre *Alice* de la première phrase, c'est-à-dire en résolvant les deux anaphores pronominales. Sachant qu'*Alice* est un prénom féminin, on obtient la phrase corrigée suivante :

(1b) C : [Alice] n'a rien mangé ce matin. Je **l**'ai LEVÉE et **l**'ai EMMENÉE à la garderie.

Dans le présent article, nous commençons par définir deux types de corrections qui requièrent la résolution des anaphores pronominales. Puis nous décrivons les algorithmes appliqués pour la résolution de ces anaphores intra- et interphrastiques afin de tenter d'effectuer chaque type de correction. Nous évaluons ensuite les résultats sur chacun de ces types, et nous montrons que l'un d'eux est de qualité suffisante pour avoir fait l'objet d'une implantation commerciale.

1.2 Conventions d'écriture des exemples

Voici les conventions graphiques utilisées dans les exemples de cet article :

- en *italique* : les antécédents potentiels selon l'algorithme ;
- [entre crochets] : l'antécédent sélectionné par l'algorithme ;
- {entre accolades} : l'antécédent réel ;
- en **gras** : les pronoms ;
- en MAJUSCULES : le mot à corriger ou corrigé par la résolution des anaphores.

Les mentions « O : » et « C : » introduisent en alternance le texte original et le texte corrigé.

2 Description des corrections recherchées

Le correcteur effectue déjà des corrections liées à des pronoms à l'intérieur d'une phrase lorsque les contraintes syntaxiques le permettent. Il reconnait ainsi les antécédents des pronoms qui sont syntaxiquement liés à leur antécédent, tels que les pronoms relatifs (ex. 2) ou les pronoms réfléchis (ex. 3). Il corrige aussi le pronom lui-même dans certains cas précis, comme dans une reprise du sujet (ex. 4). D'autre part, sans en connaitre l'antécédent, un pronom complément d'objet direct (COD) à gauche du participe passé possédant un nombre intrinsèque provoque l'accord en nombre dudit participe (ex. 5).

(2) C'est la {[*lettre*]} **que** tu as ENVOYÉ à Alice. (ENVOYÉE)

(3) {[*Alice et Élise*]} **se** sont SERRÉS l'une contre l'autre. (SERRÉES)

(4) {[*Alice*]} a-t-**IL** été sage aujourd'hui ? (ELLE)

(5) Elle **les** a VU. (VUS)

La nouveauté recherchée est d'apporter des corrections de genre ou de nombre dépendant d'un antécédent non syntaxiquement lié à un pronom, repéré dans le texte comme étant le plus probable. Dans l'exemple 5, le correcteur considère *Elle les a vus* comme sans erreur, mais, s'il parvient à déceler l'antécédent du pronom *les* et que celui-ci est féminin, il pourra alors plutôt corriger pour *vues*.

2.1 Pronoms considérés

Nous nous sommes concentrés sur les pronoms dont les référents sont généralement exprimés dans le texte, soit les pronoms personnels sujets et COD de 3^{e} personne : *il/elle/ils/elles/le/la/les/l'*. Nous avons exclu les pronoms ayant souvent des référents déictiques, tels que les pronoms de 1re et de 2^{e} personne, les pronoms démonstratifs et les pronoms possessifs. Nous avons aussi exclu les pronoms ayant des référents par défaut, tels que les pronoms indéfinis (ex. : *certains*, *tous*, etc.). Les pronoms *en, y* et *on* ne nous intéressent pas ici puisqu'ils n'entrainent aucune correction (les accords ne sont pas obligatoires et sont discutables).

2.2 Choix des corrections les plus vraisemblables

Après des essais initiaux, nous avons décidé de restreindre les corrections par résolution d'anaphores aux cas où les erreurs sont les plus vraisemblables. Par exemple, le scripteur omet plus facilement d'appliquer des accords s'ils ne sont pas marqués phonétiquement, et, comme le note (Sauvageot, 1972), l'accord en nombre est souvent silencieux (ex. 8). Une correction inaudible est donc plus vraisemblable qu'une autre qui modifie la prononciation.

Outre l'audibilité, la vraisemblance d'une erreur dépend aussi de la graphie d'un nom et de sa fréquence d'utilisation. En effet, des mots commençant par une voyelle peuvent causer des confusions de genre, particulièrement s'ils sont peu utilisés. On peut vérifier la probabilité de ces confusions par une simple recherche comparative sur Internet (ex. 6 et 7).

(6) Un anagramme ou une anagramme ? (40 200 résultats pour la recherche "*un anagramme*" et 567 000 résultats pour la recherche "*une anagramme*" le 17 février 2014 ; c'est-à-dire environ 7 % d'erreurs.)

(7) Un octave ou une octave ? (19 700 résultats pour la recherche "*un octave*" et 430 000 résultats pour la recherche "*une octave*" le 17 février 2014 ; c'est-à-dire 4 % d'erreurs.)

Notons que cette erreur est sensible à la géographie. Par exemple, on trouve 2 400 résultats sur Google pour *une autobus* sur des sites canadiens (recherche "*une autobus*" *site:ca*), alors qu'on n'en trouve que 1 280 occurrences sur des sites français (dont beaucoup de bruit de passages québécois). Nous avons donc tenu compte de l'origine du locuteur, fournie par le logiciel, pour ces corrections.

En nous basant sur ces critères de vraisemblance, nous avons décidé de tenter de corriger les accords liés aux pronoms s'ils sont inaudibles et de corriger le pronom lui-même en nombre si le nombre est inaudible, et de le corriger en genre si l'antécédent est jugé d'un genre problématique. On obtient les deux grandes classes de corrections décrites dans les points 2.3 et 2.4.

2.3 Correction en genre des accords avec les pronoms COD LES/L'

(8) O : Ces {*fleurs*} viennent de mon *jardin*. Je **les** ai CUEILLIS pour vous.
C : Ces [fleurs] viennent de mon jardin. Je **les** ai CUEILLIES pour vous.

(9) O : {*Alice*} n'a rien mangé ce *matin*. Je **l**'ai LEVÉ et **l**'ai EMMENÉ à la garderie.
C : [Alice] n'a rien mangé ce matin. Je **l**'ai LEVÉE et **l**'ai EMMENÉE à la garderie.

2.4 Correction du genre et du nombre des pronoms IL/ILS/ELLE/ELLES et LE/LA

(10) O : Je suis allée voir mes deux {*grands-mères*}. Incroyable, comme **ELLE** PARLE fort !
C : Je suis allée voir mes deux [grands-mères]. Incroyable, comme **ELLES** PARLENT fort !

(11) O : Tu peux m'aider avec cet {*anagramme*} ? **IL** n'est vraiment pas facile.
C : Tu peux m'aider avec cette [anagramme] ? **ELLE** n'est vraiment pas facile.

(12) O : Cet {*anagramme*}, je ne **LE** vois pas très clairement.
C : Cette [anagramme], je ne **LA** vois pas très clairement.

3 Système de reconnaissance de l'antécédent

Notre système doit être robuste et de couverture large puisque le correcteur est appelé à analyser n'importe quel type de texte : journaux, récits, dialogues, courriels, rapports, etc. Ces contraintes influeront sur l'ensemble du processus de sélection de l'antécédent.

La question de choisir ou non un système de détection à base de classifieurs entrainés par apprentissage automatique s'est naturellement posée. Nous étions attirés par la simplicité, la clarté et la flexibilité d'un système à base de règles qui avait déjà fait ses preuves (Trouilleux, 2002). La campagne d'évaluation de système de coréférences CoNLL-2011 a confirmé notre décision. En effet, le système de (Lee et coll., 2011) à base de règles et de *tamis*, et non à base de classification et d'apprentissage automatique, a obtenu les meilleures performances.

3.1 Survol de l'algorithme

Nous avons ainsi opté pour la même approche que (Lappin et Leass, 1994), (Mitkov 1998) et (Trouilleux, 2002), soit un algorithme à base de règles utilisant une liste d'antécédents potentiels pondérés, avec deux différences notables. Premièrement, avant de lier à un pronom la liste de ses antécédents potentiels, nous tentons de trouver l'antécédent idéal

dans la phrase même où se trouve le pronom par un système de motifs syntaxiques. Deuxièmement, alors que les algorithmes de (Lappin et Leass, 1994) et (Mitkov 1998) procèdent en deux temps, c'est-à-dire extraction des antécédents puis évaluation des couples, nous y ajoutons une troisième étape : évaluation de la probabilité de la correction selon les couples présents. Cette dernière étape est du même type que celle apportée par le système CogNIAC (Baldwin, 1997), qui sélectionne les antécédents selon le nombre de couples proposés. Voici les grandes étapes de notre système :

Pour chaque phrase :
- analyse syntaxique complète, puis sélection et extraction des antécédents *a* potentiels (point 3.3) ;
- sélection des pronoms candidats à la résolution pronominale (point 3.4).

Pour chaque pronom *p* repéré comme candidat à la résolution :
- recherche d'un antécédent local par motifs syntaxiques à partir de *p* (point 3.5) ;
- si les motifs échouent, récupération des antécédents extraits dans la phrase courante et dans les phrases précédentes puis formation des couples (*p*, *a*) ;
- évaluation des couples (*p*, *a*) selon des contraintes (point 3.6) ;
- évaluation des couples (*p*, *a*) selon des préférences (point 3.6) ;
- évaluation de la vraisemblance de la correction selon les couples (*p*, *a*) présents (point 3.7).

3.2 Contexte d'implémentation

Notre algorithme s'appuie sur plusieurs ressources fournies par le logiciel correcteur.

- Le dictionnaire. Le logiciel comporte un dictionnaire de plus de 120 000 mots, doté de traits sémantiques, morphologiques et syntaxiques. Nous savons ainsi si l'antécédent est humain ou non, s'il s'agit d'une entité comptable, entre autres ;
- L'analyseur. L'analyse est basée sur une grammaire de dépendances. Cette analyse est suffisamment robuste pour être fiable et pour retrouver des constructions complexes. Les règles dans la résolution des anaphores sont basées sur le résultat de l'analyse et nous considérons que l'analyse est toujours exacte ;
- Les régimes verbaux détaillés. L'analyseur syntaxique possède de nombreuses informations sur la formation des syntagmes verbaux. Ces informations sont très utiles pour vérifier la compatibilité sémantique d'un antécédent potentiel et d'un verbe ayant le pronom à résoudre comme complément ;
- Les cooccurrences (Charest, Brunelle, Fontaine et Pelletier, 2007). L'analyseur a été utilisé pour constituer la liste des combinaisons lexicales les plus fréquentes pour un mot selon sa fonction syntaxique. Nous évaluons grâce à cette liste la crédibilité de chaque antécédent potentiel dans la position syntaxique du pronom. Cette méthode a aussi été employée pour la traduction automatique par (Wehrli et Nerima, 2013), qui en démontrent les résultats positifs.

3.3 Sélection et extraction des antécédents potentiels

On parcourt chaque phrase pour y repérer chaque syntagme nominal (SN) potentiellement antécédent d'un pronom *p*. Notre algorithme se voulant interphrastique, il faut retenir à la volée un accès efficace à tous les antécédents potentiels au cas où on en aurait besoin, même si une phrase ne contient elle-même aucun pronom anaphorique. On repère donc les antécédents potentiels avant même de traiter les pronoms.

L'analyse syntaxique permet d'éliminer les SN qui présentent le contexte, c'est-à-dire les dates, certains circonstanciels de lieu, et autres. Ensuite, nous tentons de déterminer le degré de saillance de chaque antécédent potentiel en le pondérant, positivement ou négativement. Les règles de pondération sont autonomes, s'appliquent individuellement et sont sans ordre déterminant. Les règles les plus influentes examinent les positions syntaxiques. Elles sont hiérarchisées selon la théorie du centrage définie par (Grosz et Sidner, 1986) et décrite plus précisément par (Cornish, 2000), qui donne une liste de paramètres à considérer. Ainsi, les règles pondèrent selon la structure de la phrase (fonctions) et du SN (enchâssements), mais également selon la position du SN dans le discours (titre, début ou fin de paragraphe). Aussi, nous évaluons le degré d'actualisation du SN (déterminant, relative, etc.). Enfin, d'autres règles reconnaissent des constructions emphatiques figées comme *c'est à Paul que je parle*, et pondèrent en conséquence.

Chaque antécédent potentiel obtient ainsi un poids de saillance qui sera utile aux évaluations ultérieures. Ce poids est conservé dans une liste associée à la phrase. Par économie, un antécédent doit atteindre un seuil de poids minimal, en deçà duquel il n'est pas retenu. Nous avons fait une rapide évaluation de l'impact de ces règles dans la section 4.5.

Avant de passer à la prochaine étape, nous fusionnons certains antécédents pour en créer un nouveau afin de prévoir les cas où l'antécédent est discontinu (ex. 13). Ces antécédents ont un poids très minime, mais ils permettront d'éviter

certaines corrections indues.

(13) {*Marie*} appelle {*Jean*} et **ils** partent ensemble à la campagne.

3.4 Sélection des pronoms

L'algorithme traite les pronoms visés lors d'un deuxième parcours de la phrase. S'il existe plusieurs pronoms dans la phrase, nous les traitons un à un de gauche à droite. Deux filtres annulent la recherche d'antécédent pour un pronom :

– Le pronom est impersonnel (pronom *il* ; la reconnaissance des pronoms impersonnels a constitué un travail connexe codé au sein même de l'analyseur d'Antidote, mais non décrit par le présent article) ;
– Le pronom reprend un élément phrastique ou non nominal (pronoms *le* ou *l'* ; le pronom *la* n'est pas envisagé dans ce cadre selon notre principe de vraisemblance de l'erreur). Ces informations nous sont fournies par l'analyseur et les régimes verbaux.

Dans les deux cas, le pronom sera masculin singulier. S'il y a ambigüité (ex. 14 et 15), alors la recherche d'antécédent est tout de même lancée. L'étape d'évaluation de la correction (point 3.7) prendra en compte cet état de fait et nous ne corrigerons qu'en cas de présence d'une cooccurrence.

(14) **Il** restera toujours un perdant.

(15) Je **l**'ai vu.

3.5 Repérage d'un antécédent local par motifs syntaxiques

Une série de motifs syntaxiques est appliquée afin de tenter de trouver l'antécédent d'un pronom à l'intérieur de la phrase où il se trouve. Un « motif » est un modèle d'analyse d'une structure syntaxique où le pronom et son antécédent sont clairement liés. Ces motifs se basent sur la fonction du pronom, sur les éléments syntaxiques présents et sur l'actualisation des mots. Bien qu'euristiques, ces motifs permettent des repérages très surs et rapides. Nous en avons défini une dizaine pour les sujets et une autre dizaine pour les pronoms COD. Les exemples 16 et 17 en présentent deux. Dans le premier cas, nous avons remarqué que lorsqu'un sujet est partagé par deux verbes, et que le second verbe a pour COD un pronom, l'antécédent le plus sûr est le COD du premier verbe lorsqu'il existe. L'exemple 17 expose le cas où l'on trouve un pronom sujet dans une proposition conjonctive avant le verbe principal. Lorsque le sujet et le pronom sont de même nombre et de même genre, les deux mots sont préférablement reliés anaphoriquement.

(16) Les maitres allumèrent les {[*chandeliers*]} et **les** laissèrent bruler durant des heures.

(17) Dans ces circonstances, même si **elle** s'en défend, la {[*droite*]}, collectivement, n'a pas réellement intérêt à clarifier ses intentions.

3.6 Évaluation des couples pronom-antécédent

Lorsque les motifs échouent, nous formons la liste des antécédents potentiels. La liste des antécédents de la phrase en cours d'analyse est unie à celles des deux phrases précédentes, si elles existent, comme le préconise (Mitkov, 1998). Au passage, nous fusionnons les antécédents qui auraient été répétés dans des phrases différentes en gardant préférablement les caractéristiques de l'antécédent le plus proche, avec une augmentation du poids de saillance, le cas échéant. Chaque antécédent est pondéré selon sa distance par rapport au pronom.

Une première série d'évaluations élimine les antécédents non pertinents syntaxiquement selon le principe B de la théorie du liage (Chomsky, 1981). Ce principe dicte qu'un pronom de 3^{e} personne ne peut être lié à un syntagme présent dans son domaine de liage. L'analyse syntaxique fine de la phrase nous permet de l'appliquer très rigoureusement. Contrairement aux systèmes connus (Hobbs 1978, Lappin et Leass 1994, Dagan et Itai 1990, Baldwin 1997, Mitkov 1998, Trouilleux 2002), seuls les critères syntaxiques sont éliminatoires lorsqu'une correction est envisagée : les critères morphologiques deviennent de simples préférences.

Une deuxième série d'évaluations réajuste le poids des couples selon quatre classes de règles. Les deux premières classes sont les plus importantes et ont été les plus ardues à implémenter. Dans les deux cas, il faut simuler une phrase dans laquelle l'antécédent remplace le pronom et la soumettre à l'analyseur, puis examiner les résultats obtenus, sans pénaliser l'analyse d'un trop grand cout en temps.

– Règles sémantiques. Elles vérifient la compatibilité de l'antécédent par rapport aux éventuelles restrictions sémantiques du verbe qui porte le pronom (ex. 18). Dans l'exemple, la sémantique du verbe (*agresser*) demande un COD humain (*spectateur*) : l'antécédent non humain (*fauteuil*) peut donc être éliminé.
– Règles statistiques. Comme (Wehrli, 2013), nous utilisons les cooccurrences pour évaluer la force d'une anaphore. On peut mesurer, pour chaque antécédent, la fréquence de l'antécédent dans la position syntaxique du pronom (ex. 19).

(18a) Alors que la {[*spectatrice*]} était assise confortablement dans son *fauteuil*, le *metteur* en scène **l**'a AGRESSÉE avec des *images* effrayantes.
(18b) *Le metteur en scène agresse le *fauteuil*.
(18c) Le metteur en scène agresse la *spectatrice*.

(19) O : Pas de *problème* pour {*maman*}, *Max* **l**'a APPELÉ. <cooccurrence *appeler maman*>
C : Pas de problème pour [maman], Max **l**'a APPELÉE.

Les autres règles ne demandent qu'une comparaison, très rapide, plus simple, mais elles sont moins précises. Leur impact sur le poids de chaque couple anaphorique est plus modéré.

– Règles morphologiques. Elles vérifient la compatibilité morphologique non seulement avec le pronom, mais aussi avec tous les éléments de la phrase qui s'accordent avec le pronom. Dans l'exemple 20, on voit qu'il y a un accord non audible avec le participe passé (*trouvée*) ainsi qu'avec le premier attribut du COD (*jolie*), mais qu'il existe un accord audible avec le second (*courte*). Le poids de la morphologie sera plus fort dans ce cas.
– Règles syntaxiques. Ici, nous recherchons les parallélismes et pondérons selon les degrés de similitude. Dans l'exemple 21, on voit qu'on utilise deux fois le même verbe, avec la même distribution des arguments (nous tenons compte de la présence du semi-auxiliaire *vouloir*).

(20) C : J'ai vu la {[*robe*]} que tu as achetée à *Alice* ! Je **l**'ai TROUVÉE JOLIE, mais trop COURTE à mon gout...

(21) C : {[*Alice*]} ne voulait pas manger son {[*potage*]}, mais je l'ai forcée à y gouter. Bilan : **elle l**'a MANGÉ en deux minutes.

Pour déterminer les poids des règles, nous avons utilisé le corpus annoté ANANAS (Tutin et coll., 2000). Nous avons créé un outil de non-régression, c'est-à-dire un comparateur d'analyse pour chaque phrase du corpus, afin de repérer et de comptabiliser automatiquement les changements produits par la variation de poids d'une règle.

3.7 Évaluation de la correction et prise de décision

Dans le cas où l'antécédent ayant le poids le plus fort ne déclenche pas de correction, il est retenu : on donne le bénéfice du doute au scripteur. Bien que notre but ne soit pas de reconnaitre les antécédents des pronoms, ce résultat est tout de même présenté à l'utilisateur dans le correcteur sous la forme d'une note dans l'infobulle décrivant la nature et la fonction du pronom (figure 1). L'antécédent est aussi présenté dans l'analyse détaillée de la phrase (figure 2). Nous avons repris l'exemple 21 pour l'illustrer.

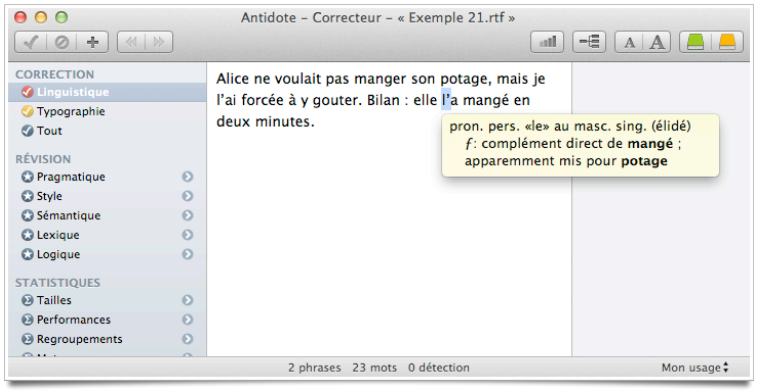


FIGURE 1 : Infobulle décrivant la nature et la fonction du pronom

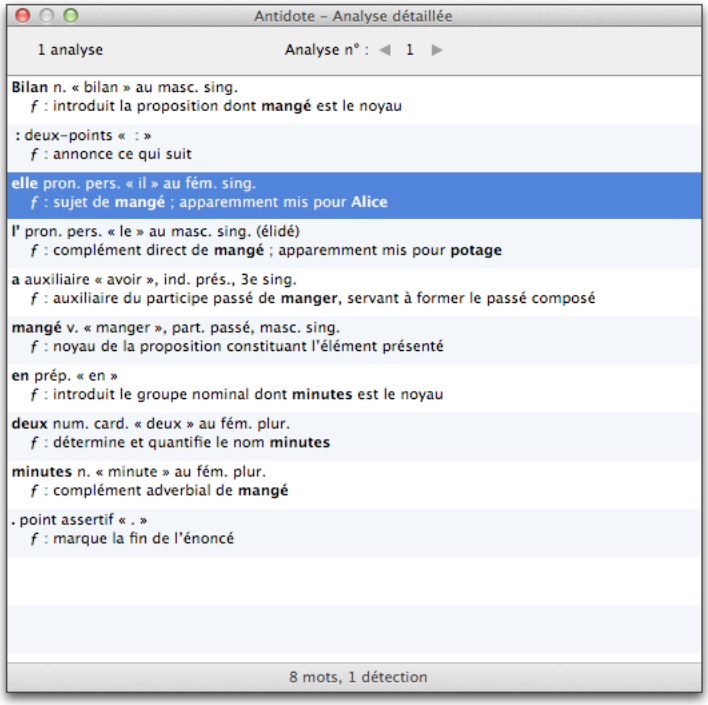


FIGURE 1 : Analyse détaillée de la phrase donnée par le logiciel de correction

Dans le cas où l'antécédent ayant le poids le plus fort déclenche une correction, nous entamons la procédure certainement la plus décisive du processus : l'évaluation de la vraisemblance de cette correction. Ce traitement prépondérant est largement euristique. Afin de filtrer le maximum de corrections indues, nous réévaluons globalement toute la situation du pronom et de son antécédent dans le texte. La sélection ne s'effectue plus selon le critère de poids du couple pronom-antécédent seul, même s'il reste important, mais selon des critères tels que la présence d'une cooccurrence entre le verbe dont le pronom dépend et l'antécédent, la différence de poids avec les autres antécédents disponibles ou la présence d'un antécédent moins fort mais sans correction. Nous vérifions aussi dans ce cadre si notre antécédent élu pourrait être lié plutôt à un autre pronom dans la phrase, ce qui éviterait la correction. Nous réévaluons aussi la correction selon l'éventualité que le pronom reprenne un élément phrastique ou soit impersonnel.

Dans le cas où nous hésitons sur une correction à apporter, car nous observons un autre antécédent possible n'apportant pas de correction ou parce que nous n'avons pas trouvé l'antécédent tout simplement, nous avons décidé d'« alerter » l'utilisateur en soulignant le pronom et en lui expliquant notre hésitation (figure 3). Le concept d'alerte est déjà utilisé pour mettre en avant des situations où l'attention de l'utilisateur est requise. Ces alertes sont soumises à un réglage qui permet à l'utilisateur de les inhiber s'il les trouve inopportunes.

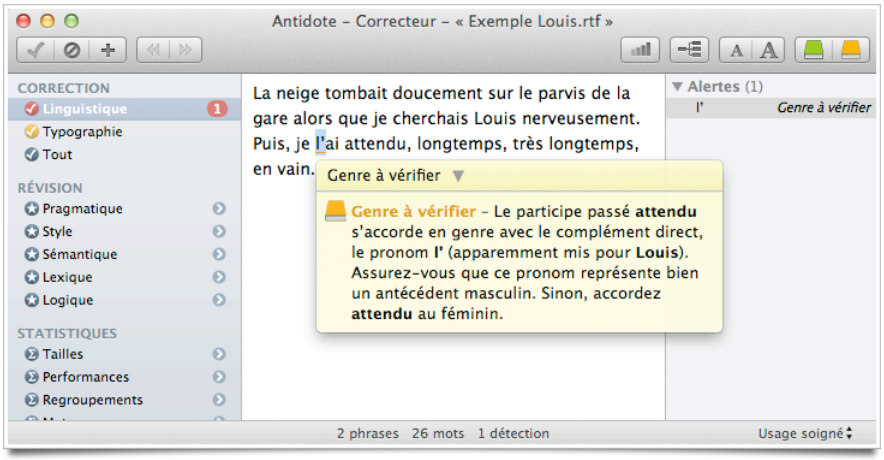


FIGURE 3 : Infobulle d'alerte

4 Résultats

4.1 Exemples de corrections obtenues

Voici quelques corrections obtenues lors de nos tests. Ces phrases sont de véritables phrases glanées sur Internet, retranscrites telles quelles. Les exemples 22 et 23 illustrent les corrections en genre (ex. 22) et en nombre (ex. 23) du pronom lui-même.

(22) O : Les {*incendies*} en Californie n'ont épargné *personne* se trouvant sur leur *chemin*. **ELLES** se sont ATTAQUÉ aux *monts* élevés comme au *fond* des *canyons*.
C : Les [incendies] en Californie n'ont épargné personne se trouvant sur leur chemin. **ILS** se sont ATTAQUÉS aux monts élevés comme au fond des canyons.

(23) O : La {*plupart*} des enfants éprouvent des difficultés à rester assis au moment des *repas* : **IL** se DANDINE, s'ASSOIT juste sur une *fesse*, se TORTILLE, se METTE à genoux…
C : La [plupart][1] des enfants éprouvent des difficultés à rester assis au moment des repas : **ILS** se DANDINENT, s'ASSOIENT juste sur une fesse, se TORTILLENT, se METTENT à genoux…

Les exemples 24, 25, 26 et 27 illustrent les corrections (toutes justifiées) effectuées sur des participes passés liés au pronom par instanciation du genre des pronoms COD *l'* et *les*. La figure 4 illustre le résultat dans le correcteur.

(24) O : J'ai l'{*épisode*}, mais je ne **l**'ai pas encore REGARDÉE.
C : J'ai l'[épisode], mais je ne **l**'ai pas encore REGARDÉ.

(25) O : D'après ces *sondages*, 18 % des *usagers* d'*appareils* numériques, ou 12 % des *foyers* américains, ont acheté une {*imprimante*} photo en 2006. Seuls *14* % d'entre eux **l**'ont ACHETÉ en kit avec un *appareil*.
C : D'après ces sondages, 18 % des usagers d'appareils numériques, ou 12 % des foyers américains, ont acheté une [imprimante] photo en 2006. Seuls 14 % d'entre eux **l**'ont ACHETÉE en kit avec un appareil.

[1] *La plupart*, bien qu'étant singulier, joue un rôle d'antécédent pluriel et doit être donc être repris par un pronom pluriel (Grevisse & Goosse).

(26) O : Soit, nous nous engageons dans une *construction* européenne rénovée qui respecte les {*Etats-Nations*} telles que l'Histoire **les** a FORGÉES.
C : Soit, nous nous engageons dans une construction européenne rénovée qui respecte les [États-Nations] tel que l'Histoire **les** a FORGÉS.

(27) O : Toutes ces {*actions*} de petite délinquance, que les *socialistes* appelaient les *incivilités*, Tony Blair **les** a APPELÉS comportements antisociaux.
C : Toutes ces [actions] de petite délinquance, que les socialistes appelaient les incivilités, Tony Blair **les** a APPELÉES comportements antisociaux.

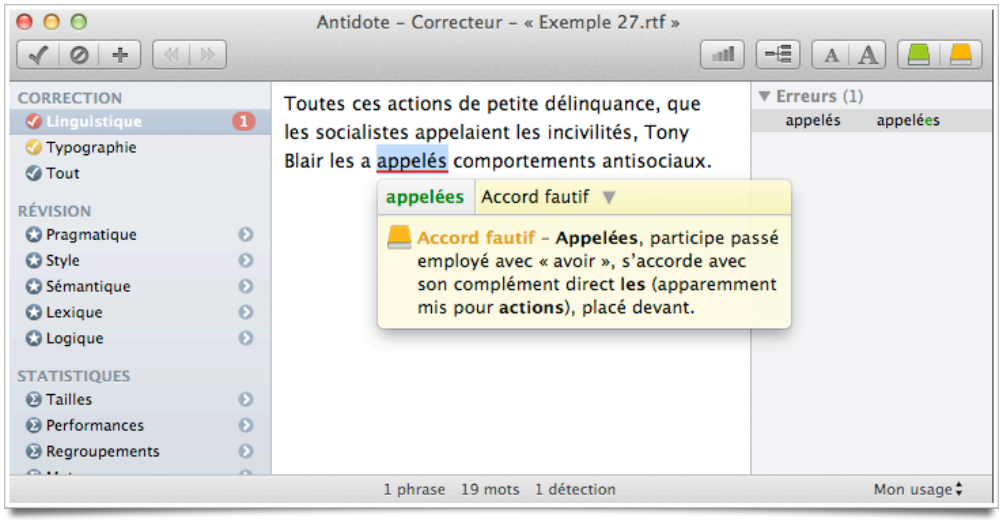


FIGURE 4 : Infobulle de correction de l'exemple 27.

4.2 Les contraintes de notre évaluation

L'évaluation classique de la résolution d'anaphores consiste à mesurer les taux de précision et de rappel de la reconnaissance de l'antécédent. Nous nous distancions de cette approche pour de multiples raisons. Tout d'abord, nous nous intéressons plutôt aux taux de précision et de rappel des corrections effectuées grâce à la résolution d'anaphores. D'autre part, l'évaluation se fait habituellement sur des textes considérés comme sans erreurs (journaux, etc.) ; dans notre cas, elle doit au contraire s'effectuer sur des textes bruts, avec de multiples erreurs. Enfin, nous ne faisons pas abstraction dans nos résultats des cas où l'erreur n'est pas due à notre système, mais plutôt à l'analyseur, qui n'aurait pas su, par exemple, faire la différence entre un pronom anaphorique et un mot d'une autre catégorie.

4.3 Évaluation de la précision

Évaluer le rappel de la correction est difficile, puisqu'il faudrait relever manuellement toutes les erreurs de référence de pronoms dans un corpus réel. Or, la configuration requise pour ce type de correction n'est pas si fréquente, et en trouver un nombre significatif serait un travail colossal. Nous avons ainsi fait le choix de n'évaluer que la précision de correction de l'algorithme, c'est-à-dire le nombre de bonnes corrections sur le nombre total de corrections.

L'évaluation de la précision nous a demandé elle aussi une réflexion, car trouver des cas de correction réels reste un défi en soi. Nous avons commencé en itérant sur un récit (un journal de voyage) augmenté d'un texte juridique, contenant plus de 20 000 phrases, afin de raffiner notre algorithme. Nous n'y avons trouvé aucun cas d'erreur qui nous intéressait. D'autre part, il nous fallait tester notre algorithme sur des textes de toutes origines, étant donné la diversité des utilisateurs du logiciel. Nous avons ainsi décidé d'utiliser le Web pour constituer un corpus plus vaste. Nous avons fait une sélection de sites selon plusieurs critères : révisés/non révisés ; caractère général/spécialisé ; présence de la 3e personne dans les pronoms personnels, objectif du site : informer/questionner/décrire/expliquer. Cette sélection comporte des textes descriptifs, informatifs, explicatifs, des biographies, des comptes-rendus de lectures, ainsi que des récits. De cette manière, nous avons extrait plus de 8 600 cas corrigés par notre algorithme. Ces 8 600 cas sont la somme des cas collectés en plus de 10 itérations sur des groupes de sites différents pour chaque itération. Notons qu'il

ne s'agit pas de 8 600 cas qui nécessitaient réellement des corrections : les cas récoltés lors de la première itération étaient très nombreux, mais étaient majoritairement des corrections injustifiées. Notre algorithme améliorant sa précision au fur et à mesure des itérations, le nombre total des corrections extraites a largement diminué, laissant une place plus importante aux corrections exactes.

Nous avons extrait finalement 152 cas lors d'une dernière itération sur de nouveaux sites ; ce sont ces cas qui nous ont servi de corpus d'évaluation, nous fournissant les chiffres du tableau 1. Sur les 152 cas, 107 corrections étaient exactes.

Types de correction	Nombre total de corrections	Nombre de corrections exactes	Précision
Correction du pronom lui-même[1]	29	18	62,0 %
Correction des éléments à accorder avec le pronom	123	95	77,2 %

TABLEAU 1 : Évaluation de la précision de la correction

L'évaluation a déterminé les types de correction à commercialiser. La partie de notre système qui corrige les éléments à accorder avec le pronom a été jugée de précision suffisante pour être commercialisée, tandis que l'autre partie devra continuer d'être améliorée.

4.4 Évaluation de l'espace mémoire et du temps de notre processus

Il est impératif de ne pas pénaliser l'utilisateur par un système de résolution trop gourmand en espace ou en temps. Après plusieurs ajustements techniques, nos résultats montrent finalement que la résolution des anaphores coute au correcteur 8 % de plus en temps et 5 % de plus en mémoire vive.

4.5 Évaluation de l'efficacité respective des critères utilisés

Nous avons fait quelques tests sur les 107 corrections exactes de notre corpus d'évaluation afin de montrer quels étaient les critères le plus importants dans la tentative de correction. Voici les trois faits les plus intéressants.

1. Sur notre corpus, 27 % des bonnes corrections sont effectuées par le biais des motifs. Donc, dans 1 cas sur 4, nous sommes en présence d'une construction relativement figée, où l'antécédent est présent dans la même phrase que le pronom. Un cas de correction indue relève des motifs (ex. 28). Il s'agit ici d'une erreur d'analyse où un syntagme nominal juxtaposé est analysé comme un élément mis en évidence (voir l'exemple 29 pour une phrase ayant la bonne analyse).

(28) O : Pourtant ce soir {*Fernando Manuel*} est humilié. Une vulgaire *histoire* de *femme*, une *portugaise* **l**'a VENDU.
C : Pourtant ce soir Fernando Manuel est humilié. Une vulgaire [histoire] de femme, une portugaise **l**'a VENDUE.

(29) La voiture, je l'ai vendue

[1] Seuls des pronoms personnels sujets ont été corrigés en nombre dans notre corpus (aucun pronom COD n'a été corrigé). Nous ignorons si cet état de fait est dû à la rareté de la correction ou à une mauvaise estimation de notre algorithme. Nous avons gardé cette question pour une évaluation future. Notons que la correction fonctionne pour des phrases avec des erreurs créées de toutes pièces.

2. En inhibant nos règles de saillance, nous perdons 31 % de nos corrections. De plus, plusieurs corrections inexactes apparaissent dans notre autre corpus de 20 000 phrases (ex. 30). La saillance a donc un impact majeur sur la correction par résolution d'anaphores.

(30) O : La {*Boule*} ne quitta pas des *yeux* l'alimentation. **IL** ÉTAIT INQUIET et ne COMPRENAIT pas pourquoi le niveau d'énergie ne montait pas.
C : La Boule ne quitta pas des [yeux] l'alimentation. **ILS** ÉTAIENT INQUIETS et ne COMPRENAIENT pas pourquoi le niveau d'énergie ne montait pas.

3. Enfin, en inhibant l'utilisation des cooccurrences, nous constatons qu'elles n'ont pas d'effet significatif sur la correction. Mais elles se montrent d'une grande efficacité dans la reconnaissance des pronoms clitiques COD pouvant reprendre un antécédent phrastique. Lorsque nous hésitons entre un antécédent nominal et un antécédent phrastique, en l'absence d'une cooccurrence, l'antécédent phrastique se révèle beaucoup plus probable. Notre résultat ne contredit pas (Wehrli et Nerima, 2013), mais cible l'aspect positif des cooccurrences dans le cadre de la résolution d'anaphores pour la correction automatique.

5 Conclusion

Cet article expose les caractéristiques de notre algorithme, lequel a la spécificité de corriger des erreurs grâce à la reconnaissance des liens anaphoriques sur un sous-ensemble de pronoms. Nous avons appliqué le principe de vraisemblance de l'erreur, et nous avons montré par nos résultats qu'il était bel et bien possible à l'heure actuelle de corriger précisément grâce à la référence pronominale intraphrastique et interphrastique. Nous avons vu qu'il est possible de mettre en place des solutions mélangeant plusieurs stratégies, telles que les motifs syntaxiques et la traditionnelle liste d'antécédents pondérés selon la méthode de (Lappin et Leass, 94), actualisée et précisée avec l'ajout des règles liées aux cooccurrences, sans que cela soit couteux en temps ou en mémoire.

Le code est maintenant testé et éprouvé quotidiennement par un demi-million d'utilisateurs. Quelques rares utilisateurs nous ont signalé une correction indue ou, au contraire, un silence, directement reliés à notre système. Notons qu'il est difficile de juger de la qualité du système sur ces requêtes, puisqu'il est rarissime qu'un utilisateur nous écrive pour nous féliciter d'une correction exacte. Pour notre part, en tout cas, nos propres textes ont bénéficié plusieurs fois de corrections reliées aux anaphores.

Nous avons depuis continué notre travail, et nous avons mis en place de façon connexe un système de motifs apportant des corrections nouvelles pour les pronoms de 2e personne ainsi que pour la correction des cas de pronoms reprenant des éléments phrastiques. Nous continuons aussi d'améliorer l'analyse syntaxique, renforçant de ce fait la qualité des corrections par les anaphores. Nous projetons de continuer notre travail sur la correction des pronoms eux-mêmes par l'ajout de nouvelles méthodes innovantes.

Remerciements

Merci à Mala F. Bergevin pour ses conseils linguistiques toujours avisés, et à Guy Lapalme pour ses suggestions concernant l'écriture de cet article.

Références

BALDWIN B. (1997). CogNIAC: A High Precision Pronoun Resolution Engine. Technical report, University of Pennsylvania.

CHAREST, S., BRUNELLE, E., FONTAINE, J., PELLETIER, B. (2007). Élaboration automatique d'un dictionnaire de cooccurrences grand public. Actes de *TALN 2007*, 283-292.

CHOMSKY, N. (1981). *Lectures on Government and Binding*, Dordrecht, Foris.

CORNISH, F. (2000). L'accessibilité cognitive des référents, le centrage d'attention et la structuration du discours : une vue d'ensemble. *Verbum*, Vol. XXII, no 1, 7-30.

DAGAN, I., ITAI, A. (1990). Automatic Processig of Large Corpora for Resolution of Anaphora References. Acte de 13th Conference on Computational Linguistics (COLING'90), 3, 330-332

GREVISSE, M., GOOSSE, A. (2007). *Le Bon Usage : grammaire française*, 14e éd., Bruxelles : De Boeck Duculot, 2008.

GROSZ, B., SIDNER, C. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3), 175-204.

HOBBS, J. (1978). Resolving Pronoun Reference. *Lingua* 44, 311-338.

LEE, H., PEIRSMAN, Y., CHANG, A., CHAMBERS, N., SURDEANU, M., JUFRAFSKY, D. (2011). Stanford's Multi-Pass Sieve Corefernce Resolution System at the CoNLL-2011 Shared Task. Acte de *CoNLL-2011 : Shared Task,* June, 73.

LAPPIN, S., LEASS, H. (1994). An Algorithm for Pronominal Anaphora Resolution. *Computational Linguistics* 20(4), 535-561.

MITKOV, R. (1998). Robust pronoun resolution with limited knowledge. Acte de *Annual Meeting of the ACL*, 869-875.

SAUVAGEOT, A. (1972). *Analyse du français parlé*. Paris : Hachette.

TROUILLEUX, F. (2002). A Rule-based Pronoun Resolution System for French. Acte de *4th Discourse Anaphora and Anaphor Resolution Colloquium*.

TUTIN A., TROUILLEUX F., CLOUZOT C., GAUSSIER E., ZAENEN A., RAYOT S., ANTONIADIS G. (2000). Annotating a large corpus with anaphoric links. Actes de *Discourse Anaphora and Anaphor Resolution (DAARC 2000)*.

WEHRLI, E., NERIMA, L. (2013). Collocations and anaphora resolution in machine translation. Acte de *Workshop on Multi-Word Units in Machine Translation and Translation Technologie*.

Peut-on bien chunker avec de mauvaises étiquettes POS ?

Isabelle Tellier[1,2], Iris Eshkol-Taravella[3,4] , Yoann Dupont[3], Ilaine Wang[3]
(1) université Paris 3 – Sorbonne Nouvelle (2) Lattice, UMR 8094,
(3) université d'Orléans (4) LLL, UMR 7270

iris.eshkol@univ-orleans.fr, isabelle.tellier@univ-paris3.fr,
yoann.dupont@etu.univ-paris3.fr, i.wang@u-paris10.fr

Résumé. Dans cet article, nous testons deux approches distinctes pour chunker un corpus oral transcrit, en cherchant à minimiser les étapes de correction manuelle. Nous ré-utilisons tout d'abord un chunker appris sur des données écrites, puis nous tentons de ré-apprendre un chunker spécifique de l'oral à partir de données annotées et corrigées manuellement, mais en faible quantité. L'objectif est d'atteindre les meilleurs résultats possibles pour le chunker en se passant autant que possible de la correction manuelle des étiquettes POS. Nos expériences montrent qu'il est possible d'apprendre un nouveau chunker performant pour l'oral à partir d'un corpus de référence annoté de petite taille, sans intervention sur les étiquettes POS.

Abstract. In this paper, we test two distinct approaches to chunk transcribed oral data, trying to minimize the phases of manual correction. First, we use an existing chunker, learned from written texts, then we try to learn a new specific chunker from a small amount of manually corrected labeled oral data. The purpose is to reach the best possible results for the chunker with as few manual corrections of the POS labels as possible. Our experiments show that it is possible to learn a new effective chunker for oral data from a labeled reference corpus of small size, without any manual correction of POS labels

Mots-clés : chunker, étiquetage POS, apprentissage automatique, corpus oral, disfluences
Keywords: chunker, POS labeling, machine learning, oral corpus, disfluencies

1 Introduction

Nous nous intéressons dans cet article au processus de *segmentation* de textes en chunks, c'est-à-dire en constituants continus non-récursifs (Abney, 1991). La tâche de chunking vise en effet à identifier la structure syntaxique superficielle d'un énoncé, c'est-à-dire à reconnaître ses constituants minimaux, sans pour autant spécifier leur structure interne ni leur fonction syntaxique. Elle s'appuie sur un étiquetage morpho-syntaxique (ou POS) préalable, donnant ainsi lieu à une séquence d'annotations successives.

Plusieurs stratégies sont possibles pour construire un chunker. L'apprentissage automatique supervisé est particulièrement performant sur cette tâche (Sha et Pereira, 2003), surtout si l'étiquetage POS sur lequel il repose est de bonne qualité. Mais le résultat d'un processus d'apprentissage n'est pas toujours adapté à des textes différant sensiblement de ceux ayant servi à apprendre. Nous supposons être dans la situation suivante : nous disposons d'un étiqueteur POS et d'un chunker appris à partir d'une assez grande quantité de données annotées (les données sources), homogènes en termes de style. Nous souhaitons maintenant chunker des textes nouveaux (les données cibles), initialement non annotés, présentant de grandes différences de style avec les données sources. En particulier, l'annotation POS produite sur les données cibles par le modèle résultant de l'apprentissage sur les données sources n'est pas de bonne qualité, mais nous ne souhaitons pas consacrer du temps à apprendre un nouvel étiqueteur morpho-syntaxique spécifique pour le corpus cible. Dans ce cas, est-il utile de corriger manuellement les étiquettes POS du

corpus cible pour faciliter la tâche au chunker qui opère sur elles, ou vaut-il mieux se concentrer sur le seul niveau du chunking ? C'est la principale question que nous abordons dans cet article.

Dans le cas exploré ici, les données sources sont des textes journalistiques, et les données cibles des transcriptions de l'oral. L'oral se caractérise par des phénomènes linguistiques qui lui sont propres, regroupés sous l'appellation générale de *disfluences,* qui compliquent son annotation et son chunking. L'intérêt du chunking de l'oral est pourtant indéniable : il représente un degré d'analyse adapté pour certains énoncés où l'on constate des libertés prises par rapport à une syntaxe standard. Il a par exemple été montré que les chunks sont le lieu de réalisation privilégié des réparations à l'oral (Blanche-Benveniste C., 1997 : 47).

Notre objectif est donc de chunker le mieux possible nos données orales cibles, en minimisant l'intervention manuelle. Nous souhaitons notamment voir s'il est possible d'acquérir un chunker de l'oral de bonne qualité à partir de peu de données annotées, sans pour autant apprendre un étiqueteur POS spécifique. Apprendre un chunker est en effet moins coûteux qu'apprendre un étiqueteur POS car la variabilité des données servant d'appui (les étiquettes POS dans un cas, les mots dans l'autre) est moindre. Une situation similaire peut survenir dans d'autres contextes, par exemple pour adapter un reconnaisseur d'entités nommées (lui aussi largement fondé sur un étiquetage POS préalable) acquis sur des textes écrits à des données orales. Et la même problématique d'adaptation se pose aussi si, au lieu que ce soit la modalité (écrit/oral) qui change entre les données sources et cibles, c'est leur domaine, leur genre, voire leur langue.

L'article suit la structure suivante. Tout d'abord, nous évoquons la tâche de chunking, ses spécificités dans le cas de l'oral ainsi que les corpus source et cible à notre disposition : le corpus annoté de textes écrits (French Treebank) de Paris 7 et un extrait du corpus oral transcrit ESLO 1 (section 2). Nous décrivons ensuite (en section 3) les différents chunkers utilisés : ils proviennent tous de la même technique d'apprentissage automatique supervisée, mais partant de données annotées différentes. Nous exposons enfin dans la dernière partie (la section 4) les résultats de diverses stratégies utilisées pour chunker les données orales transcrites, nécessitant différents degrés de corrections manuelles.

2 La tâche et les données

2.1 Chunking des données orales transcrites

Les chunkers, aussi appelés « shallow parsers », sont bien adaptés aux données orales transcrites, dont les énoncés ne sont pas souvent « finalisés ». Deux problèmes majeurs se posent aux outils annotant l'oral : les disfluences, qui rompent la linéarité du discours, et le manque de ponctuation dans les transcriptions. Pour (Dister, 2007), les disfluences sont les « marques typiques des énoncés en cours d'élaboration » qui « constituent un piétinement sur l'axe syntagmatique de l'énoncé et […] nécessitent d'être prises en compte par le système d'étiquetage. ». Les disfluences typiques sont les suivantes (extraits du corpus ESLO, décrit plus loin) :

- les hésitations : *madame euh comment vous faîtes une omelette*
- les faux-départs : *il va y avoir encore des encore mais*
- les répétitions : *le le*
- les autocorrections : *juste après le la fin du premier cycle*
- les reformulations : *on fait ce que l'on appelle un carton c'est-à-dire le le ce dessin-là agrandi*
- les amorces : *vous v- vous êtes in- institutrice*
- etc.

Elles représentent un vrai problème pour l'analyse automatique de l'oral (Adda-Decker et al., 2003, Antoine et al., 2003, Benzitoun, 2004, Valli et Véronis 1999) et réduisent considérablement les performances des outils construits pour l'écrit standard. Nos propres expériences confirmeront ce constat (cf. section 4.1). La notion de phrase, essentiellement graphique, a rapidement été abandonnée par les linguistes qui s'intéressent à l'oral ; les transcriptions ne sont donc en général pas ponctuées pour éviter l'anticipation de l'interprétation (Blanche-Benveniste, Jeanjean, 1987).

Il existe des solutions spécifiques pour le chunking du français transcrit :

- (Blanc et al., 2008, 2010) ont essayé d'annoter un corpus oral français en « super-chunks » (chunks contenant les multi-mots complexes), en appliquant des cascades de transducteurs utilisant des ressources lexicales et syntaxiques. Le processus est fondé sur une étape de prétraitement des données consistant dans le reformatage et l'étiquetage des disfluences. Une approche similaire a été adoptée par (Valli et Véronis 1999) pour l'étiquetage morphosyntaxique de l'oral.
- (Antoine et al., 2008) ont proposé une autre stratégie incluant une étape de post-correction pour traiter les erreurs liées aux disfluences.

Suite à (Blanche-Benveniste, 2005), nous considérons quant à nous que les phénomènes de disfluences doivent être inclus *dans* l'analyse linguistique, même s'ils exigent des traitements spécifiques. Pour faire face aux données réelles et éviter les programmes *ad hoc* écrits à la main, nous privilégions les techniques issues de l'apprentissage automatique.

2.2 Le French TreeBank (FTB) et ses étiquettes

Le premier corpus dont nous devons tenir compte, notamment parce qu'il a fixé les jeux d'étiquettes que nous utilisons (aussi bien au niveau des POS qu'à celui des chunks), est le FTB (le French TreeBank)[1]. Il s'agit d'un corpus de phrases écrites syntaxiquement analysées qui peut être facilement transformé en phrases annotées en POS et en chunks (Abeillé et al., 2003). Le jeu réduit de 30 étiquettes POS est décrit dans (Crabbé et Candito 2008). Les six types de chunks extraits de ces données, avec les étiquettes POS correspondant à leur tête, sont les suivants :

- les groupes nominaux ou *NP* (incluant *CLO, CLR, CLS, NC, NPP, PRO, PROREL, PROWH* : notons que les pronoms sont ici considérés comme des chunks nominaux autonomes et pas inclus dans les noyaux verbaux) ;
- les groupes verbaux ou *VN*, incluant les formes interrogatives, infinitives et modales (*V, VIMP, VINF, VPP, VPR, VS*) ;
- les groupes prépositionnels ou *PP*, incluant les groupes nominaux introduits par une préposition (*P, P+D, P+PRO*) ;
- les groupes adjectivaux ou *AP*, incluant les éventuels adverbes modifieurs d'adjectifs (*ADJ, ADJWH*) ;
- les groupes adverbiaux ou *AdP*, incluant les modifieurs de phrases (*ADV, ADVWH, I*) ;
- les groupes de conjonction ou *CONJ* (*CC, CS*).

2.3 ESLO 1

Le deuxième corpus utilisé est un tout petit extrait du corpus oral transcrit ESLO 1 (Enquête Sociolinguistique d'Orléans)[2] (Eshkol-Taravella et al. 2012) constitué de 8093 mots correspondant à 852 tours de parole (3 entretiens face-à-face). Les conventions de transcription dans ESLO respectent deux principes : l'adoption de l'orthographe standard et le non-recours à la ponctuation de l'écrit. Les marques typographiques comme le point, la virgule, le point d'exclamation ou encore la majuscule en début d'énoncé sont absentes. La segmentation a été faite soit sur une unité intuitive de type « groupe de souffle » repérée par le transcripteur humain, soit sur un « tour de parole », défini simplement par le changement de locuteur. Les données traitées dans le cadre de ce travail correspondent au corpus transcrit brut non annoté et non lemmatisé.

3 Etiqueteur et chunkers utilisés

3.1 SEM, un étiqueteur-chunker appris sur le FTB

L'outil d'annotation utilisé dans un premier temps est SEM[3] (Tellier et al., 2012), un segmenteur-étiqueteur capable d'enchaîner plusieurs annotations successives. SEM est spécialisé dans l'analyse des textes écrits, puisqu'il a été appris

[1] http://www.llf.cnrs.fr/Gens/Abeille/French-Treebank-fr.php

[2] http://eslo.tge-adonis.fr/

[3] http://www.lattice.cnrs.fr/sites/itellier/SEM.html

uniquement à partir du FTB, et ses étiquettes sont donc celles présentées précédemment. Il permet soit de chunker un texte déjà annoté en POS, soit d'enchaîner « annotation POS + chunks » sur du texte brut. Nous exploiterons par la suite ces deux usages distincts.

SEM a été appris à l'aide d'un CRF (Conditional Random Fields) linéaire (Lafferty et al. 2001), implémenté dans le logiciel Wapiti[4] (Lavergne et al. 2010). Pour l'étiquetage en POS, SEM utilise une ressource extérieure : le LeFFF (Lexique des Formes Fléchies du Français) (Sagot 2010) intégré dans les données sous la forme d'attributs booléens. Pour le chunker, le modèle CRF s'appuie à la fois sur l'étiquetage POS et sur les tokens initiaux.

Le découpage en chunks est traduit par une annotation qui suit le format standard *BIO* (*B* pour Beginning, *I* pour In, *O* pour Out). Avec SEM, chaque mot (ou token) du corpus reçoit donc, outre son étiquette POS, une étiquette qui est la concaténation du type de chunk dont il fait partie et d'une étiquette (*B* ou *I*) qui indique la position qu'il y occupe.

3.2 Technique d'apprentissage de nouveaux chunkers

Nous ne chercherons pas à apprendre un nouvel étiqueteur POS spécifique de l'oral mais plutôt, dans certaines expériences à apprendre un nouveau chunker à partir de données orales annotées à la fois en POS et en chunks. Pour apprendre ce nouveau chunker (en fait, il y en aura plusieurs, suivant la nature des étiquettes utilisées), nous utiliserons, comme cela avait été fait pour apprendre SEM, des CRF linéaires.

Les CRF sont des modèles graphiques probabilistes non dirigés, discriminants et particulièrement efficaces pour la prédiction d'étiquettes. Dans le cas des modèles linéaires, ils cherchent la meilleure séquence d'étiquettes *y* à associer à la séquence de données d'entrée *x*, en maximisant une probabilité P(y|x). La probabilité P(y|x) s'exprime dans un CRF par une combinaison pondérée (les poids étant les paramètres de l'apprentissage) de fonctions caractéristiques (ou features), qui caractérisent des configurations locales de données et d'étiquettes. Pour définir l'ensemble des features de son modèle, l'utilisateur d'un programme comme Wapiti spécifie des patrons (ou templates) : sortes d'expressions régulières pouvant faire intervenir n'importe quelle propriété des données d'entrée, et une (dans le cas des patrons unigrammes) ou deux (patrons bigrammes) étiquettes successives. Les patrons seront instanciés sur l'ensemble d'apprentissage, constitué de couples (x,y), en autant de features que de positions où ils peuvent s'appliquer.

Dans le cas de l'apprentissage d'un chunker, les données d'entrée *x* sont constituées des séquences de tokens (ou mots) du texte et des étiquettes POS associées, la suite des étiquettes cibles *y* est constituée des différents types de chunks associés à *B* ou *I*. Les patrons que nous utiliserons pour apprendre ce(s) nouveau(x) chunker(s) ont été copiés sur ceux utilisés pour l'apprentissage de SEM, et seront toujours les mêmes pour chaque expérience. Ils figurent dans la Table 1 :

Attribut	Fenêtre sur x	Type de feature sur y
token	[-2, 0]	unigramme
POS	[-2, 1]	unigramme et bigramme
Couple de POS	{-2, 0} et {-1, 0}	unigramme

TABLE 1 : spécification des patrons (templates) définissant les features des modèles CRF de chunking

4 Deux séries d'expériences

Nous décrivons dans cette section les deux séries d'expériences réalisées avec le corpus oral transcrit et les résultats obtenus. La Table 2 montre l'annotation du même exemple extrait d'ESLO 1 par différents processus, qui incluent (en gras : colonnes **III**, **IV** et **V**) ou non (colonne **II**) une phase de correction manuelle. Les corrections manuelles ont toutes été assurées par une unique personne experte. Les différentes colonnes de ce tableau serviront soit de données d'entrée

[4] http://wapiti.limsi.fr

soit de données de référence à nos différentes expériences. Leurs contenus seront décrits en détail au fur et à mesure que nous les présenterons. Pour nos évaluations, deux chunks seront considérés comme égaux lorsqu'ils partagent exactement les mêmes frontières et le même type. Nous évaluerons les résultats du chunking avec la micro-average des F-mesures des différents chunks (moyenne des F-mesures de ces chunks pondérées par leurs effectifs) et leur macro-average (moyenne sans pondération des F-mesures). Notons que sur le FTB, en validation croisée à 10 plis, SEM a été évalué avec une exactitude 97,33% pour l'étiquetage en POS, une micro-average de 97,53 et une macro-average de 90,4 pour le chunker. Les Tables 3 et 5 (en fin d'article) donnent respectivement les proportions des différents types de chunks et la synthèse de l'ensemble de nos résultats.

I	**II**	**III**	**IV**	**V**
Tokens	POS proposés par SEM	**POS corrigés à la main**	**Chunks « type FTB » corrects**	**Chunks adaptés à l'oral corrects**
euh	*DET*	***I***	***AdP-B***	***IntP-B***
l-	*DET*	***UNKNOWN***	***AdP-B***	***UNK-B***
dans	*P*	***P***	***PP-B***	***PP-B***
ma	*DET*	***DET***	***PP-I***	***PP-I***
classe	*NC*	***NC***	***PP-I***	***PP-I***

TABLE 2 : les différentes données d'entrée/de référence utilisées

4.1 Première approche : utilisation d'un chunker appris sur l'écrit

4.1.1 Utilisation directe de SEM

Le premier test consiste à appliquer SEM, sans aucune adaptation ni ré-apprentissage, sur les données transcrites cibles de l'oral. SEM est utilisé sur le texte brut, et produit en cascade l'étiquetage en POS et celui correspondant au chunking. Dans la Table 2, cela correspond à prendre comme données d'entrée pour le POS la colonne **I** (les tokens), et comme données d'entrée pour le chunker les colonnes **I** et **II** (les étiquettes POS fournies par SEM sur ESLO 1).

Pour évaluer la qualité du chunking produit par SEM sur l'oral, il faut constituer un corpus de référence en corrigeant l'annotation en chunks proposée par SEM sur l'extrait d'ESLO 1, avec les étiquettes qu'il utilise (colonne **IV** dans la Table 2). Découper en chunks la transcription de l'oral pose des problèmes spécifiques, à cause, entre autres, des disfluences. Nous explicitons ici les choix faits pour cette correction manuelle.

L'exemple de l'énoncé annoté dans la Table 2 (*euh l- dans ma classe*) montre bien le type de difficultés rencontrées. Les *euh* d'hésitation, ne pouvant pas être une tête de chunk, constituent des chunks adverbiaux (*AdP)*. Cette décision concerne également les interjections (d'étiquette POS *I*) comme dans l'exemple ci-dessous :

(on/CLS)NP (peut/V)VN (commencer/VINF)VN ***(bon/I)AdP*** *(alors/I)AdP*

Les faux départs et les amorces (comme *l-* dans l'exemple de la Table 2), quand ils sont impossibles à interpréter, font également partie de chunks adverbiaux (*AdP)*. Dans les cas où une interprétation est possible, l'annotation se fait selon le contexte. Dans l'exemple :

(vous/PRO)NP (êtesV)VN ***(in-/NC)NP*** *(institutrice/NC)NP*

l'amorce *in-* semble correspondre exactement au début du mot suivant *institutrice*, elle est donc annotée en tant que nom commun (*NC*) et forme par conséquent un chunk nominal autonome (*NP*). Dans l'exemple suivant :

(chez/P vous/PRO)PP (chez/P ***v-/PRO)PP***

la répétition de la même préposition *chez* et l'équivalence entre l'amorce *v-* et le début du pronom *vous*, laisse supposer qu'il s'agit de la répétition du même groupe prépositionnel.

Les répétitions de type « faits de parole », font partie des disfluences de l'oral (contrairement aux « faits de langue » où la répétition est due à la syntaxe (Henry, 2005)). Deux possibilités se présentent alors pour le chunking :

- Si l'élément répété est la tête du groupe syntaxique, il est nécessaire de distinguer deux chunks, car un chunk ne peut pas contenir deux têtes distinctes :

  (et/CC)CONJ (et/CC)CONJ *(elle/CLS)NP (me/CLO)NP (disait/V)VN*

- Si la répétition ne porte pas sur une tête, les deux éléments appartiennent au même chunk :

  (la/DET la/DET *belle/ADJ jeune/ADJ fille/NC)NP*

Le chunking produit par SEM sans aucune adaptation est évalué relativement à cette référence avec une micro-précision de 77,24 et une macro-précision de 76. Plus de 20 points de F-mesure en moyenne (micro-average) sont donc perdus en appliquant un programme appris avec des textes écrits sources sur des données transcrites de l'oral. Ce mauvais résultat est le point de départ de différentes tentatives d'amélioration. L'objectif des expériences qui suivent est de corriger le minimum de données manuellement pour améliorer au maximum les performances du chunker.

4.1.2 Utilisation de SEM après correction de l'étiquetage POS

Le chunking précédent était appliqué en cascade après un étiquetage POS du corpus qui était lui-même sans doute médiocre. La première idée pour améliorer le chunking est donc de corriger manuellement l'étiquetage POS de l'oral avant de lui appliquer la phase de chunking. Ce processus a permis par la même occasion d'évaluer la qualité de l'étiquetage POS de SEM sur l'oral : son exactitude atteint 80,98%, soit 17% de moins environ que sur des données similaires à celles qui ont servi à apprendre. La fonction « chunker seul » de SEM peut ensuite s'appliquer sur le corpus avec des étiquettes POS corrigées à la main (les colonnes **I** et **III**).

Pour corriger les étiquettes POS, certaines conventions ont été adoptées concernant les disfluences de l'oral (voir les colonnes **II** : les étiquettes POS annotées par SEM et **III** : les étiquettes POS corrigées selon les conventions établies). Les faux départs et les amorces (comme *l-* dans l'exemple de la Table 2) ont reçu une étiquette *(UNKNOWN)* qui correspond aux mots étrangers et aux néologismes dans le FTB. Les marqueurs discursifs ainsi que les *euh* d'hésitation ont été étiquetés en tant qu'interjection (*I*). C'est, parmi les étiquettes disponibles dans SEM, celle qui correspond le mieux à ces unités caractéristiques de l'oral.

La correction des erreurs de l'étiquetage POS porte surtout sur les différences entre l'écrit et l'oral. Par exemple, la forme *bon* est utilisée en tant qu'adjectif dans 99% des cas dans le FTB, alors qu'elle est beaucoup plus fréquente dans le corpus oral en tant qu'interjection (83%).

La nouvelle micro-average du chunker est maintenant de 87,74 alors que sa nouvelle macro-average est de 88,43. Ces résultats sont en quelque sorte à mi-chemin des précédents : à peu près la moitié des erreurs de chunking sur l'oral peut donc être imputée à des erreurs d'étiquetage POS.

4.2 Deuxième approche : Apprentissage d'un chunker spécifique de l'oral

La deuxième approche consiste à apprendre un nouveau chunker à partir du seul corpus extrait d'ESLO 1, en tenant compte autant que possible des spécificités de l'oral. Nous avons choisi de ne pas ré-appprendre un étiqueteur POS spécifique sur les données cibles (ni à en appliquer un autre que SEM), pour nous concentrer sur la phase de chunking. Tant qu'à ré-apprendre un nouveau chunker, nous en avons aussi profité pour définir un jeu de chunks adapté.

4.2.1 Modification des étiquettes de chunks

Pour tenir compte des spécificités de l'oral, nous avons choisi d'ajouter deux types de chunks nouveaux qui lui sont propres (voir la colonne **V** du Table 2). La liste des chunks a ainsi été élargie par deux nouveaux venus :

- chunk *UNKNOWN*

L'étiquette *UNKNOWN* existe dans le jeu d'étiquettes POS du FTB, où elle est attribuée aux mots étrangers. Nous l'avons utilisée aussi pour désigner les chunks correspondant aux erreurs de transcriptions, aux faux départs ou aux amorces dont l'interprétation est impossible. Dans notre exemple de la Table 2, la forme *l-* est difficile à comprendre.

S'agit-il d'un pronom, d'un déterminant ou d'une amorce ? L'étiquette *UNKNOWN*, déjà choisie pour cette forme au niveau POS, est donc étendue dans ce cas au chunk.

- chunk d'interjection (*IntP*)

Nous avons déjà signalé le problème que posent les marqueurs discursifs et les *euh* d'hésitation qui ont été classés, faute d'avoir une autre étiquette davantage adaptée dans SEM, dans les chunks adverbiaux. L'ajout d'un nouveau chunk *IntP* (chunk interjection) destiné à accueillir tous ces phénomènes, résout (au moins partiellement) ce problème :

(des/DET idées/NC laïques/ADJ)NP (quoi/I) IntP

Cependant, lorsque les interjections se trouvent à l'intérieur d'un groupe syntaxique, ils s'intègrent dans le chunk correspondant :

- *(l'/DET école/NC **euh/I** publique/ADJ)NP*
- *(des/DET **hm/I** inconvénients/NC)NP*

Dans les deux exemples ci-dessus, le *euh* d'hésitation et l'interjection *hm* appartiennent à un chunk nominal.

Ce nouvel étiquetage en chunks a été manuellement validé sur nos données ESLO (colonne **V** de la Table 2), et constitue la nouvelle référence grâce à laquelle nous allons à la fois apprendre et évaluer notre nouveau chunker.

4.2.2 Apprentissage et test avec les étiquettes POS corrigées

La première expérience consiste à apprendre un chunker à partir des données cibles annotées en POS corrigées (la colonne **III** de la Table 2) et des chunks adaptés à l'oral (la colonne **V**). Un protocole de validation croisée à 10 plis a été utilisé pour évaluer la qualité du chunker ainsi obtenu, quand il est appliqué à des données de nouveau parfaitement annotées en POS. La micro-average des F-mesures atteint alors 96,65 alors que leur macro-average vaut 96,08. Les résultats se sont donc significativement améliorés, et rejoignent ceux qui avaient été constatés pour SEM sur FTB.

Si on observe de plus près les F-mesures des différents types de chunks, en comparaison avec les expériences précédentes, on constate une forte progression de l'annotation des chunks adverbiaux (*AdP*). Ces chunks sont très nombreux dans notre corpus au cours des premières expériences, car ils regroupent les adverbes, les marqueurs discursifs, les *euh* d'hésitation et les interjections. L'introduction d'un nouveau chunk *(IntP)* annotant ces différents phénomènes (sauf les adverbes) a considérablement réduit le nombre de chunks adverbiaux dans le corpus de référence, ce qui modifie significativement leur F-mesure. Lors de ces premières expériences, la F-mesure du chunk *(AdP)* varie entre 58,14 (avec les POS non corrigées) et 71,87 (avec les POS corrigées). Désormais, la F-mesure atteint 95,76 pour le chunk *(AdP)* et 99,4 pour le chunk *(IntP)*. L'apprentissage a donc bien réussi à distinguer les deux types de chunks.

Les erreurs constatées concernent souvent des « exceptions » aux règles générales. C'est le cas des verbes qui forment d'habitude un chunk verbal *(VN)* sauf quand ils suivent une préposition. Ainsi, dans l'exemple ci-dessous, le verbe est annoté comme la tête d'un chunk verbal :

(à/P)PP (me/CLR)B-NP (marier/VINF)B-VN

alors qu'il fait partie ici d'un chunk prépositionnel (*PP*) :

(à/P me/CLR marier/VINF)PP

Les cas où les interjections et les marqueurs formant généralement un chunk *(IntP)* sont inclus dans un autre chunk posent aussi problème. Le chunker appris propose :

*(l'/DET école/NC)NP **(euh/I) IntP** (publique/ADJ)AP*

à la place de :

(l'/DET école/NC ***euh/I*** *publique/ADJ)****NP***

Enfin, en cas de répétition de deux étiquettes morphosyntaxiques, le chunker inclut parfois les deux mots dans le même chunk, violant ainsi la contrainte qui voudrait que chaque chunk ne devrait contenir qu'une seule tête. Il annote ainsi :

(et/CC parce_que/CS)CONJ

(ils/CLS)NP (réfléchissaient/V pensaient/V)VN (beaucoup/ADV)AdP

à la place de

(et/CC)CONJ (parce_que/CS)CONJ

(ils/CLS)NP (réfléchissaient/V)VN (pensaient/V)VN (beaucoup/ADV)AdP

Mais les très bons résultats de ce nouveau chunker ne sont atteints que sur des données qui ont elles-mêmes reçu un étiquetage POS parfaitement correct. Or, aucun étiqueteur POS de l'oral n'ayant été appris, notre nouveau chunker risque de voir ses performances se dégrader significativement en situation réelle, c'est-à-dire avec de mauvaises étiquettes POS. Pour quantifier ce problème et essayer d'y remédier, nous avons mené deux nouvelles expériences qui ne font pas l'hypothèse de disposer d'un étiquetage POS corrigé lors de la phase *d'utilisation* du chunker.

4.2.3 Apprentissage avec les étiquettes POS corrigées, test sur les étiquettes non corrigées

La deuxième expérience de cette série vise à évaluer la dégradation de performance subie quand le chunker appris sur des étiquettes POS corrigées (colonnes **III** et **V**) est utilisé sur des données avec des étiquettes POS non corrigées (colonne **II**). Etant donné le faible volume de données dont nous disposons, nous avons pour cela reconduit l'expérience précédente en validation croisée à 10 plis, en prenant soin lors de chaque étape de respecter le protocole suivant :

- l'apprentissage est réalisé à l'aide des colonnes **I**, **III** et **V**
- le chunker appris est appliqué en test sur les colonnes **I** et **II**
- le résultat obtenu est comparé à la colonne de référence **V**

Nous obtenons ainsi une micro-average des F-mesure de 73,81, et une macro-average de 59,62, ce qui représente une grosse dégradation (cf. les détails des valeurs des différents chunks dans la Table 3). Les performances sont particulièrement mauvaises pour le nouveau type de chunk *IntP*, car très peu d'étiquettes POS *I* sont correctement attribuées par SEM dans ESLO 1. En effet, dans le FTB, les seules interjections présentes correspondent à des phrases d'un seul mot suivi d'une ponctuation. Or ESLO 1 ne contient pas de ponctuation, et cet indice n'aide donc en rien le chunker. La plupart des interjections de ESLO 1 comme *bon, bien, enfin, alors, etc*. sont étiquetées par SEM comme des adverbes ou des adjectifs lors de l'étiquetage POS. Le nouveau chunker appris les rattache alors à un chunk adverbial plutôt qu'à un chunk IntP. Un seul chunk *IntP* a été reconnu lors de cette expérience, et cela semble-t-il de façon quasiment « fortuite ». Les représentants du nouveau chunk *(UNKNOWN)* n'ont pas non plus été identifiés, ce qui s'explique naturellement par le fait que SEM n'a pas attribué l'étiquette POS *UNKNOWN* là où notre correction manuelle l'avait fait (sur les disfluences en particulier). Le problème mentionné précédemment et concernant la forme *bon* persiste également dans les résultats de ce test. Ayant une étiquette POS *ADJ*, *bon* est étiqueté en tant que chunk adjectival *(AP)* et non comme chunk interjection *(IntP)*. Dans l'exemple suivant (où la dernière colonne donne la proposition du nouveau chunker, tandis que l'avant-dernière donne la bonne étiquette), les deux unités *bon* et *alors* reçoivent une mauvaise étiquette de chunk :

on	*CLS*	*B-NP*	*B-NP*
peut	*V*	*B-VN*	*B-VN*
commencer	*VINF*	*B-VN*	*B-VN*
bon	*ADJ*	*B-IntP*	*B-AP*
alors	*ADV*	*B-IntP*	*B-AdP*

L'absence de correction des POS cause donc ici des erreurs prévisibles de chunking, surtout pour les nouveaux types de chunks qui s'appuient sur des propriétés de l'oral que les étiquettes POS non corrigées de SEM ne prennent pas en compte. Il reste à voir si un chunker appris directement sur des étiquettes POS non corrigées se comporterait mieux.

4.2.4 Apprentissage et test avec les étiquettes POS non corrigées

La dernière expérience vise à apprendre le chunker de l'oral en se servant uniquement des étiquettes POS fournies par SEM, sans aucune correction (ni en apprentissage ni en test) sur ces POS. Cette fois, notre validation croisée emploie donc les colonnes **I**, **II** et **V** de la Table 2, en cherchant à obtenir la dernière de ces colonnes à partir des deux autres. L'objectif de cette dernière expérience est de voir s'il est possible d'apprendre un bon chunker en se fondant sur des étiquettes POS médiocres. Existe-t-il des régularités dans les erreurs au niveau morpho-syntaxique dont l'apprentissage pourrait tirer parti ? Pourrait-on donc se passer d'une correction manuelle de l'étiquetage POS (et d'un ré-apprentissage d'un étiqueteur POS de l'oral) pour obtenir tout de même *in fine* un chunker de l'oral correct ? C'est tout l'enjeu de cet ultime test.

Nous obtenons dans cette expérience une micro-average de 88,84, et une macro-average de 81,76, soit des résultats (comme on pouvait s'y attendre) intermédiaires entre les deux précédents (cf. les détails dans la Table 4). Cette fois, on constate que les chunks *(IntP)* sont très bien reconnus (plus de 93 de F-mesure), alors que SEM substitue à l'étiquette POS correcte *I* des étiquettes assez variées (typiquement ADV, ADJ, NC et V). Mais les interjections sont à la fois fréquentes et assez peu variées dans notre corpus de l'oral (*euh, hm, oui, non, etc.*) et celles présentes dans l'ensemble d'apprentissage suffisent apparemment au chunker appris (qui a aussi accès aux mots ou tokens et pas uniquement aux POS) à les identifier. Ainsi, l'exemple précédent reçoit cette fois l'étiquetage :

on	*CLS*	*B-NP*	*B-NP*
peut	*V*	*B-VN*	*B-VN*
commencer	*VINF*	*B-VN*	*B-VN*
bon	***ADJ***	***B-IntP***	***B-IntP***
alors	*ADV*	*B-IntP*	*B-AdP*

La forme *bon* est étiquetée ici correctement au niveau des chunks *(B-IntP)* malgré une erreur d'étiquetage POS où elle est reconnue comme un adjectif. Dans le corpus d'apprentissage, ce mot est le plus souvent employé comme marqueur discursif, ce qui facilite sa désambiguïsation. Les unités *oui, non,* aussi très fréquentes dans le corpus, reçoivent maintenant aussi une bonne étiquette de chunk, quelle que soit leur étiquette POS.

Sur le chunk *(UNKNOWN)*, le nouveau chunker obtient une bonne précision (92,86%) :

vous	*DET*	*B-NP*	*B-NP*
êtes	*NC*	*B-VN*	*B-VN*
in-	***ADJ***	***B-UNKNOWN***	***B-UNKNOWN***
institutrice	*NC*	*B-NP*	*B-AdP*
n-	***ADV***	***B-UNKNOWN***	***B-UNKNOWN***
peut-être	*VINF*	*B-AdP*	*B-AdP*
non	***ADV***	***B-IntP***	***B-IntP***

mais un mauvais rappel (18,57%). Cela tient sans doute au fait que les chunks inconnus peuvent parfois correspondre à des mots connus mais employés dans un mauvais contexte, comme dans l'exemple suivant :

euh	***V***	***B-IntP***	***B-IntP***
les	***DET***	***B-UNKNOWN***	***B-NP***
dans	*P*	*B-PP*	*B-PP*
ma	*DET*	*I-PP*	*I-PP*
classe	*NC*	*I-PP*	*I-PP*

En outre, les amorces présentent une bien plus grande variabilité que les interjections ; toutes ne peuvent pas être présentes dans l'ensemble d'apprentissage et l'accès aux tokens ne suffit donc pas à compenser le mauvais étiquetage POS. Il ne semble pas y avoir de règle évidente quant aux chunks *(UNKNOWN)* bien identifiés. L'hypothèse la plus probable est que SEM a reconnu uniquement les mots qu'il a déjà vus dans son ensemble d'apprentissage. On pourrait sans doute largement améliorer les capacités de notre nouveau chunker à reconnaître les amorces en lui donnant accès à certaines propriétés des tokens : dans ESLO 1, les amorces sont en effet systématiquement terminées par un tiret - : ajouter cette propriété aux attributs pris en compte dans les features devrait permettre de les identifier bien plus sûrement que par leur contexte. Mais nous voulions utiliser le même ensemble de features (copiées sur celles utilisées pour l'apprentissage de SEM) pour toutes nos expériences, pour ne pas biaiser les comparaisons.

Type de chunk	PP	AdP	VN	AP	NP	CONJ	UNKNOWN	IntP
FTP	33,66%	7,23%	17,11%	2,21%	32,95%	6,61%	N/A	N/A
ESLO, chunks FTB	11,84%	11,93%	25,43%	2,66%	39,79%	8,36%	N/A	N/A
ESLO, chunks oral	9,39%	9,98%	21,81%	2,38%	33,7%	8,15%	1,14%	13,45%

TABLE 3 : proportions des différents types de chunks dans les différents corpus

Les détails des résultats obtenus sur les différents types de chunks pour les deux dernières expériences sont présentés dans la Table 4.

Type de chunk	Expérience 4			Expérience 5		
	Précision	Rappel	F-mesure	Précision	Rappel	F-mesure
AP	50,73%	71,23%	59,26	71,76%	64,38%	67,87
AdP	55,9%	79,48%	65,64	83,78%	85,83%	84,79
CONJ	89,42%	89,42%	89,42	89,8%	91,42%	90,6
IntP	33,33%	0,12%	0,24	95,82%	91,87%	93,8
NP	81,16%	85,34%	83,2	91,93%	90,6%	91,26
PP	71,99%	81,55%	76,48	81,57%	82,41%	81,99
UNKNOWN	N/A	N/A	N/A	92,86%	18,57%	30,95
VN	78,13%	87,23%	82,43	89,75%	90,89%	90,32

TABLE 4 : Résultats des différents types de chunks dans les deux dernières expériences

La synthèse des résultats de l'ensemble de nos expériences est présentée dans la Table 5.

Expériences	Première approche : utilisation d'un chunker appris sur l'écrit (référence : la colonne **IV**)		Deuxième approche : Apprentissage d'un chunker spécifique de l'oral (référence : la colonne **V**)		
Evaluation	POS non corrigées	POS corrigées	POS corrigées	Apprentissage sur POS corrigés, test sur POS non corrigés	POS non corrigés
Exactitude des POS (%)	*80,98*	*100*	*100*	*80,98*	*80,98*
Micro-average	*77,24*	*87,74*	*96,65*	*73,81*	*88,84*
Macro-average	*76*	*88,43*	*96,08*	*59,62*	*81,76*

TABLE 5 : Synthèse des résultats des micro et macro-averages des F-mesures dans l'ensemble de nos expériences

5 Conclusion

Tout d'abord, notre première série d'expériences montre qu'un étiqueteur morphosyntaxique associé à un chunker, tous deux appris sur un corpus source écrit fait environ 17% d'erreurs supplémentaires en POS, et 20% en chunking, sur des données cibles orales transcrites. Cet écart important justifie de trouver des stratégies d'adaptation ou de contournement

pour traiter les corpus oraux. Comme l'erreur en chunking n'est pas beaucoup plus importante que l'erreur en POS, la solution de corriger les POS apparaît *a priori* comme la plus « naturelle ». Cette correction manuelle des POS améliore le résultat du chunking de 10 points de F-mesure en moyenne, mais reste 10 points en dessous des performances moyennes du chunker sur l'écrit. Même avec un étiquetage POS parfait, l'écart entre l'écrit et l'oral en matière de chunking se mesure avec ces 10 points d'écart en moyenne.

Corriger directement les étiquettes de chunks apparaît donc comme la suite logique de cette approche. Nous avons pour cela choisi de coller aux propriétés de l'oral plutôt que de chercher à faire entrer à tout prix les données orales dans le cadre défini pour l'écrit, d'où le choix des deux nouveaux types de chunks introduits. Ce faisant, nous n'avons pas choisi la facilité car la tâche de chunking devient plus complexe (il faut désormais discriminer parmi huit types de chunks au lieu de six). Pour l'apprentissage automatique d'un nouveau chunker spécifique de l'oral, le pari a été fait de se consacrer au seul niveau des chunks, pour lequel un petit nombre de données d'apprentissage peut suffire.

Les trois expériences de la deuxième approche permettent de caractériser assez finement l'apport des étiquettes POS à la phase de chunking. En présence d'étiquettes POS correctes et cohérentes avec les chunks (première expérience), l'apprentissage automatique joue parfaitement son rôle, et permet d'apprendre un chunker d'aussi bonne qualité que celui qui avait été appris sur l'écrit avec beaucoup plus de données. Il n'y a donc pas de malédiction propre à l'oral en matière de chunking : même les disfluences peuvent y être bien traitées, à condition de disposer d'exemples de référence, même en quantité restreinte. En revanche, un tel chunker dépend fortement des étiquettes POS sur lesquelles il s'appuie : l'absence de correction manuelle (deuxième expérience de la série) fait chuter ses performances. Il n'est donc pas réellement exploitable en conditions réelles : en effet, tant qu'à corriger les étiquettes POS, autant ré-apprendre dans ce cas un étiqueteur POS de l'oral…

La dernière expérience est la plus prometteuse : elle montre qu'on peut apprendre un chunker spécifique de l'oral (y compris pour la reconnaissance des interjections par exemple) d'assez bonne qualité, en s'appuyant uniquement sur un petit nombre de données annotées, qui plus est avec des étiquettes POS médiocres (et non adaptées à l'oral). Les erreurs du POS ont bien été *compensées* par l'apprentissage du chunker, qui fait en moyenne moins d'erreurs de chunking qu'il n'y a d'erreurs d'étiquetage POS. Les mots, même en petites quantités, permettent cette compensation, et sans doute aussi le fait que les erreurs de POS sont suffisamment « régulières » pour que le chunker puisse les « rectifier ».

L'apprentissage automatique d'un chunker spécifique de l'oral semble donc pouvoir assez bien se passer d'un étiquetage POS correct. Il est intéressant de constater que les résultats obtenus pour le chunker dans la dernière expérience sont très proches de ceux de la deuxième expérience de la première approche, c'est-à-dire en appliquant SEM sur des étiquettes POS corrigées manuellement. La différence est que le nouveau chunker obtenu avec la dernière expérience est applicable sans plus de correction manuelle sur de nouvelles données orales, ce qui n'est pas le cas de ce que proposait l'autre expérience. Ainsi, tant qu'à corriger des données, il vaut semble-t-il mieux s'attaquer aux données qui servent à l'apprentissage (les nouveaux chunks dans la dernière expérience) qu'aux données qui servent de support à un programme déjà appris (les POS dans l'expérience 2).

Il reste bien sûr à confirmer que le même genre de démarche peut être valable dans d'autres contextes, par exemple pour d'autres tâches (la reconnaissance des entités nommées pouvant se substituer à celle de chunking), ou en changeant la variation écrit/oral par une autre, comme un changement de domaine ou de type d'écriture (les tweets pourraient par exemple remplacer l'oral). Le fait que l'apprentissage direct d'un nouvel étiqueteur focalisé sur une tâche cible est préférable à une séquence d'apprentissages intermédiaires avait par ailleurs déjà été constaté (Eshkol et al., 2010). Le caractère cumulatif des erreurs n'est donc pas une fatalité : il semble qu'on puisse réussir une tâche de « haut niveau » en s'appuyant sur des informations de « niveau inférieur » de qualité médiocre par apprentissage automatique, du moment que la correction des erreurs d'un niveau à un autre suive une certaine régularité.

Références

ABNEY S. (1991). Parsing by chunks. In R. Berwick, R. Abney, and C. Tenny, editors, *Principle-based Parsing*. Kluwer Academic Publisher.

ABEILLE A., CLEMENT L., et TOUSSENEL F. (2003). Building a treebank for french. In A. Abeillé, editor, *Treebanks*. Kluwer, Dordrecht.

ADDA-DECKER M., HABERT B., BARRAS C., ADDA G., BOULA DE MAREÜIL P., PAROUBEK P. (2003). A disfluency study for cleaning spontaneous speech automatic transcripts and improving speech language models. In Proceedings of *Isca tutorial and research workshop on disfluency in spontaneous speech (diss'03),* 67-70.

ANTOINE J-Y., GOULIAN J., VILLANEAU J. (2003). Quand le TAL robuste s'attaque au langage parlé : analyse incrémentale pour la compréhension de la parole spontanée. Actes de *TALN 2003*, 25-34.

ANTOINE J-Y., MOKRANE A., et FRIBURGER N. (2008) Automatic rich annotation of large corpus of conversational transcribed speech: the chunking task of the epac project. In Proceedings of *LREC'2008*.

BENZITOUN C. (2004). L'annotation syntaxique de corpus oraux constitue-t-elle un problème spécifique ? Actes de *RÉCITAL*.

BLANC O., CONSTANT M., DISTER A. et WATRIN P. (2008). Corpus oraux et chunking. Actes de *Journées d'étude sur la parole (JEP),* Avignon, France.

BLANC O., CONSTANT M., DISTER A. ET WATRIN P. (2010). Partial parsing of spontaneous spoken French. In Proceedings of *7th International Conference on Language Resources and Evaluation (LREC'10)*.

BLANCHE-BENVENISTE C. (2005). Les aspects dynamiques de la composition sémantique de l'oral. *Sémantique et corpus.* A. Condamines (dir.), Londres, Hermes, 40-73.

BLANCHE-BENVENISTE C., JEANJEAN C. (1987). *Le français parlé, transcription et édition.* Paris, Didier Erudition.

BLANCHE-BENVENISTE C. (1997). *Approches de la langue parlée en français*. Paris, Ophrys.

BLANCHE-BENVENISTE C. (2000). Transcription de l'oral et morphologie. *Romania Una et diversa, Philologische Studien für Theodor Berchem* (Gille M. et Kiesler R. Eds). Tübingen : Gunter Narr, 61-74.

CONSTANT M., TELLIER I. (2012) Evaluating the impact of external lexical ressources unto a crf-based multiword segmenter and part-of-speech tagger. In Proceedings of *LREC 2012.*

CRABBE B, CANDITO M (2008). Expériences d'analyse syntaxique du français. Actes de *Traitement Automatique des Langues Naturelles (TALN 2008*), Avignon.

DISTER A. (2007). *De la transcription à l'étiquetage morphosyntaxique. Le cas de la banque de données textuelle orale VALIBEL*. Thèse de Doctorat, Université de Louvain.

ESHKOL I., TELLIER I, TAALAB S., BILLOT S., (2010). Étiqueter un corpus oral par apprentissage automatique à l'aide de connaissances linguistiques. Actes de *10es Journées Internationales d'analyse statistique des données textuelles (JADT 2010)*.

ESHKOL-TARAVELLA I., BAUDE O., MAUREL D., HRIBA L., DUGUA C., TELLIER I., (2012) Un grand corpus oral « disponible » : le corpus d'Orléans 1968-2012. Dans *Ressources linguistiques libres*, *TAL* 52, n° 3, 17-46.

HENRY S. (2005). Quelles répétitions à l'oral ? Esquisse d'une typologie, G. Williams (Éd.), *La Linguistique de corpus*, Rennes, Presses universitaires de Rennes, 81-92.

LAVERGNE T, CAPPE O, ET YVON F. (2010). Practical very large scale CRFs. In Proceedings of *ACL'2010*, 504–513.

LAFFERTY J, MCCALLUM A, ET PEREIRA F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of *ICML 2001*, 282–289.

SAGOT B. (2010). The Lefff, a freely available, accurate and large-coverage lexicon for French. In Proceedings of *7th International Conference on Language Resources and Evaluation (LREC'10)*.

SHA F, PEREIRA P. (2003). Shallow parsing with conditional random fields. In Proceedings of HLT-NAACL, 213–220.

TELLIER I., DUCHIER D., ESHKOL I., COURMET A., MARTINET M. (2012), Apprentissage automatique d'un chunker pour le français, Actes de *Traitement Automatique des Langues Naturelles (TALN 2012*).

TELLIER I., ESHKOL I., TAALAB S., PROST J-P. (2010). POS-tagging for Oral Texts with CRF and Category Decomposition. *Research in Computer Science,* special issue *: Natural Language Processing and its Applications*, 79-90.

VALLI A., VERONIS J. (1999). Etiquetage grammatical des corpus de parole : problèmes et perspectives. L'oral spontané. *Revue Française de Linguistique Appliquée* IV-2, 113-133.

Normalisation de textes par analogie: le cas des mots inconnus

Marion Baranes[1,2] Benoît Sagot[2]
(1) viavoo, 92100 Boulogne Billancourt
(2) Alpage, INRIA & Université Paris-Diderot, 75013 Paris
{prenom.nom}@inria.fr

Résumé. Dans cet article, nous proposons et évaluons un système permettant d'améliorer la qualité d'un texte bruité notamment par des erreurs orthographiques. Ce système a vocation à être intégré à une architecture complète d'extraction d'information, et a pour objectif d'améliorer les résultats d'une telle tâche. Pour chaque mot qui est inconnu d'un lexique de référence et qui n'est ni une entité nommée ni une création lexicale, notre système cherche à proposer une ou plusieurs normalisations possibles (une normalisation valide étant un mot connu dont le lemme est le même que celui de la forme orthographiquement correcte). Pour ce faire, ce système utilise des techniques de correction automatique lexicale par règle qui reposent sur un système d'induction de règles par analogie.

Abstract. Analogy-based Text Normalization : the case of unknowns words. In this paper, we describe and evaluate a system for improving the quality of noisy texts containing non-word errors. It is meant to be integrated into a full information extraction architecture, and aims at improving its results. For each word unknown to a reference lexicon which is neither a named entity nor a neologism, our system suggests one or several normalization candidates (any known word which has the same lemma as the spell-corrected form is a valid candidate). For this purpose, we use an analogy-based approach for acquiring normalisation rules and use them in the same way as lexical spelling correction rules.

Mots-clés : normalisation textuelle, correction orthographique, analogie.

Keywords: Text normalization, Spell checking, Analogy.

1 Introduction

La qualité orthographique d'un texte peut avoir de nombreux impacts sur les analyses faites en traitement automatique de la langue. Les outils pour le français par exemple sont très souvent conçus pour traiter du français standard. La qualité de leurs analyses et leurs résultats se retrouvent par conséquent assez dégradés dès qu'il s'agit de traiter des textes plus bruités. Le système présenté ici tend à améliorer la qualité d'un texte donné pour une tâche d'extraction d'informations. L'outil effectuant cette tâche réalise ses analyses en s'appuyant sur des séquences de mots représentés par leurs lemmes. Notre objectif est d'améliorer la qualité des textes qu'il traite avant leur passage dans l'outil en les normalisant.

La normalisation est une tâche qui a pour but de proposer, pour chaque mot considéré comme « fautif », sa forme normativement correcte ou une forme qui lui est flexionnellement liée. Nous considérons qu'une forme est « fautive » si son orthographe peut gêner une analyse ultérieure. Notre travail se concentre sur la normalisation des fautes lexicales, qui correspondent à des mots inconnus de notre lexique de référence, ici, le L*efff* (Sagot, 2010), en laissant de côté pour l'instant les fautes grammaticales non flexionnelles, où un mot est remplacé par un autre mot connu du lexique et de lemme différent (ex : *sont/font* ou *et/est*). Enfin, nous voulons normaliser un texte et non le corriger, traiter les fautes grammaticales flexionnelles (fautes d'accord ou de flexion) n'est donc pas pertinent. Bien que la majorité des formes que nous considérons ici comme fautives correspondent à des fautes lexicales, il peut aussi s'agir d'une forme réduite volontairement (cf. *changemt* pour *changement*). De manière générale, si nous voulons normaliser une forme « fautive », deux possibilités s'offrent à nous. Nous pouvons soit tenter de la normaliser comme le ferait un outil de correction automatique, en proposant sa forme normativement correcte (ex : *téléphonnez* → *téléphonez*), soit la remplacer par une forme qui partage le même lemme que sa forme attendue (ex : *téléphonnez* → *téléphoner* ou *téléphoniez*). Ce dernier rattachement ne provoquera aucun bruit puisque nous cherchons ici à améliorer un texte pour qu'il puisse être analysé par une tâche ultérieure qui, comme indiqué plus haut, s'appuie non pas sur les tokens mais sur les lemmes.

Notre objectif est donc de transformer un texte bruité en remplaçant les mots « fautifs » qu'il contient par leur correction idéale ou par une forme fléchie du lemme qui était attendu. Pour ce faire, nous faisons le postulat qu'en utilisant

des techniques de correction automatique nous obtiendrons un bon système de normalisation. Notre système, inspiré des systèmes par analogie, fonctionne ainsi à l'aide de règles de correction apprises et pondérées automatiquement. Naturellement, une tokenisation préalable est nécessaire. De plus, certains mots « fautifs » ne pourront pas être corrigés par analogie, et ne sont parfois pas véritablement fautifs (les abréviations, par exemple). Nous appliquons donc un certain nombre de traitements préalables avant de mettre en œuvre notre système de normalisation par analogie proprement dit.

Comme ce système est inspiré des systèmes de correction automatique lexicale et des systèmes utilisant l'analogie, nous commencerons par dresser un état de l'art de ces notions à la section 2. Puis, nous détaillerons le système présenté et les différents modules qui le composent à la section 3, y compris les traitements préalables à son application. Enfin, nous évaluons notre système à la section 4.

2 État de l'art

Le système de normalisation présenté ici s'inspire de techniques de correction automatique et plus particulièrement de la correction des fautes lexicales. Les travaux sur la correction lexicale ont débutés dans les années 1960 (Blair, 1960; Damerau, 1964). Au fil du temps et des avancées technologiques, ces travaux ont beaucoup évolué (Kukich, 1992; Mitton, 1996, 2010). L'état de l'art proposé ci-dessous, n'a donc pas pour objectif d'être exaustif mais plutôt de donner un aperçu des techniques utilisées. Les premiers systèmes proposés se limitaient au mot à corriger sans prendre en compte leurs mots avoisinants. Ils fonctionnaient principalement à base de règles typographiques et de distance d'édition (Damerau, 1964; Kernighan *et al.*, 1990) ou avec un système de vérification dans le lexique plus tolérant (Oflazer, 1996). Puis, le contexte a commencé à être pris en compte. Pour ce faire, certains travaux se sont appuyés sur des modèles de langue n-grammes de mots (Brill et Moore, 2000; Carlson et Fette, 2007; Park et Levy, 2011), phonétiques (Toutanova et Moore, 2002), ou encore sur des modèles de langue n-grammes qui combinaient ses deux caractéristiques (Boyd, 2009). Ces n-grammes étant, par la suite, souvent associés à d'autres paramètres tels que la catégorie grammaticale, la transcription phonétique ou encore la longueur du mot à corriger. Des études différentes ont préféré s'appuyer sur des mesures distributionnelles. C'est par exemple le cas de Suignard et Kerroua (2013), qui se sont appuyés sur les travaux de Li *et al.* (2006). Enfin d'autres approches, comme celle proposée par Yvon (2011) s'appuient plutôt sur des modèles probabilistes.

Le choix des techniques utilisées est souvent guidé par l'objectif visé par le correcteur. Un correcteur qui cherche à corriger des fautes grammaticales en plus des fautes lexicales (comme ceux proposés par Carlson et Fette (2007) et Yvon (2011)) devra automatiquement être plus robuste, la détection d'erreurs étant plus complexe à réaliser et la génération de candidats de correction plus risquée. Par ailleurs, si le correcteur en question doit traiter des textes plus dégradés (SMS, forum, blogs, etc.), les approches varieront aussi. On pourra, dans ce cas, ajouter des lexiques plus spécifiques au type de fautes à traiter (Guimier de Neef *et al.*, 2007; Seddah *et al.*, 2012), prendre en compte la phonétique des mots en phonétisant le texte à traiter (Kobus *et al.*, 2008), utiliser un système de classification (Han et Baldwin, 2011) ou encore procéder à un apprentissage par alignement (Beaufort *et al.*, 2010).

Dans ce travail, nous voulons corriger des mots sans prendre en compte le contexte dans lequel ils apparaissent. C'est pour cette raison que nous nous sommes intéressés aux systèmes par règles. Ces systèmes reposent souvent sur la notion de distance d'édition introduite par Damerau (1964) puis Levenshtein (1966). Cette notion met en oeuvre quatre types de règles (l'insertion et la suppression d'une lettre, la substitution d'une lettre par une autre, l'inversion de deux lettres consécutives). L'idée est de s'appuyer sur ces opérations, que l'on peut donc considérer comme des règles de correction, pour passer d'un mot mal orthographié à sa forme attendue. Cette méthode a été reprise et améliorée par de nombreux travaux. Des règles spécifiques aux fautes de proximité clavier sont proposées (Sagot et Boullier, 2008). Des connaissances linguistiques ont également été intégrées. Par exemple, Véronis (1988) et Sagot et Boullier (2008) traitent les fautes dites phonétiques (ex : remplacement de o par eau). Le contexte dans lequel s'applique la règle peut être pris en compte (Kernighan *et al.*, 1990). Par ailleurs, la pondération de ces règles ne se fait plus systématiquement en fonction du nombre d'opérations effectuées sur le mot. Le choix du poids d'une correction se fait, manuellement ou non, en fonction de l'opération effectuée et des lettres concernées par cette opération (Véronis, 1988; Mitton, 1996; Sagot et Boullier, 2008).

Ces règles s'appuient majoritairement soit sur des règles simples que l'on tente d'appliquer sur toutes les lettres d'un mot inconnu, soit sur des règles plus spécifiques (proximité clavier, phonétique) souvent listées manuellement. Notre objectif est d'acquérir automatiquement ces règles et de les pondérer en fonction de la fréquence de la faute correspondante. Pour cela, nous nous inspirons de techniques d'apprentissage par analogie pour induire des règles de correction.

L'analogie a déjà été employée pour plusieurs tâches relevant du traitement automatique des langues. Elle peut ainsi être utilisée pour des tâches de traduction automatique (Lepage et Denoual, 2005) ou de recherche d'informations (Moreau

et al., 2007), pour regrouper les mots d'un lexique en famille morphologique (Hathout, 2010) ou encore pour l'analyse morphologique (Stroppa et Yvon, 2005; Lavallée et Langlais, 2011). Cette notion permet d'établir un rapport de proportionnalité entre des unités linguistiques. Formellement, elle peut être représentée ainsi : « $a : b :: c : d$ », ce qui signifie « a est à b ce que c est à d ». Le couple a, b entretient ainsi une relation d'analogie avec c, d. En morphologie, par exemple, on pourra construire des analogies comme « *tutoiement* : *tutoyer* :: *vouvoiement* : *vouvoyer* ». Nous ne définirons pas ici dans le détail la notion d'analogie. Le lecteur intéressé pourra se référer aux travaux de (Lepage, 1998, 2000), de (Stroppa et Yvon, 2006) ou encore de (Lavallée et Langlais, 2009). Dans ce travail, l'analogie nous permettra d'analyser et de rattacher un mot inconnu dont l'orthographe est considérée comme « fautive » à son orthographe attendue par analogie avec des couples (forme fautive, forme corrigée) connues.

3 Système proposé

Les formes que nous pourrions considérer comme « fautives » sont nombreuses et de types très distincts. Toutefois, les prétraitements effectués au préalable de ce module nous permettront d'écarter à terme les fautes spécifiques aux corpus de type SMS ou les formes trop éloignées de leurs orthographes normativement correctes (ex : *tjs* ou *pk*). Ainsi la majorité des fautes restantes seront souvent dues à des fautes de proximité clavier (ex : *znalise*), de duplication de lettres (ex : *anallyse*), phonétiques (ex : *analize*) ou encore provenant d'un mécanisme de réduction volontaire de mot (ex : *pvoir*). Ces fautes semblent donc pouvoir être rapprochées d'autres fautes possibles résultant de mécanismes similaires. Nous faisons l'hypothèse, dans cet article, que la majorité de ces fautes résultent au plus d'une erreur lourde, comme montré par Damerau (1964), éventuellement associée à des fautes « légères d'accentuation, et que ces différents types de fautes peuvent être corrigées par des règles apprises automatiquement par analogie formelle. Par exemple, si nous avons déjà vu au préalable et extrait par généralisation appropriée la règle de correction permettant de passer de « *engagemt* » à « *engagement* » nous devrions être capable de prédire l'orthographe correcte de « *changemt* » (cf Figure 1).

	forme fautive		forme correcte
::	engagemt	:	engagement
::	changemt	:	?

$\rightarrow$

	forme fautive		forme correcte
::	engagemt	:	engagement
::	changemt	:	changement

FIGURE 1 – Correction du mot *changemt* par analogie

Le système proposé dans cet article utilise donc des techniques de correction par règles afin de normaliser les textes que nous souhaitons traiter. Avant de nous intéresser plus particulièrement au fonctionnement de ce dernier, nous décrirons rapidement, à la section 3.1, les traitements mis en oeuvre au préalable de notre module. Puis nous détaillerons, dans les sections 3.2, 3.3 et 3.4, l'apprentissage de nos différentes règles de correction pondérées, leur combinaison les unes avec les autres et l'application de ces dernières sur les mots considérés comme « fautifs » par notre système. L'objectif étant de proposer des candidats de correction pondérés aux formes inconnues du L*eff* que nous considérons comme potentiellement « fautives ».

3.1 Traitements préalables

Avant d'appliquer notre module de normalisation, nous effectuons plusieurs traitements tels que la tokenisation de notre texte, la détection de motifs particuliers ou encore la correction de formes spécifiques à certains types de corpus (« *tkt* » qui signifie « *ne t'inquiète pas* » par exemple). Nous réalisons ces traitements à l'aide de la chaîne SXPipe (Sagot et Boullier, 2008). Plus précisément, nous sommes partis de sa version utilisée dans le cadre du projet EDyLex (Sagot *et al.*, 2013), qui cherche à pallier certains problèmes que pose l'incomplétude lexicale et qui a notamment pour but de détecter les néologismes. Nous y avons ajouté des modules de détection de certains phénomènes non encore traités, afin de ne pas modifier des formes qui n'auraient pas de raisons de l'être. C'est par exemple le cas des autocensures (ex : *m*#%@*, *p***n*), des onomatopées (ex : *hahaha*), des interjections (ex : *hum*), des étirements (ex : *nooooon*) ou des décompositions (ex : *dé-plo-rable*). Enfin, nous avons ajouté des modules de correction/normalisation déterministe pour traiter certaines fautes fréquentes qu'une approche reposant sur des règles induites par analogie ne pourra pas traiter. Nous utilisons pour cela un lexique de corrections [1] et une courte liste d'autres formes non standard [2], notamment des abréviations (ex : *ds* ou

1. Ce lexique peut-être mis en place et complété manuellement mais nous n'utilisons actuellement que les 924 fautes courantes figurant dans la liste de correction automatique de la wikipédia (cf : `http://fr.wikipedia.org/wiki/Wikip\%C3\%A9dia:Liste_de_fautes_d\%27orthographe_courantes`)
2. L'orthographe de ces mots étant souvent voulue, nous préférons parler de formes non standard plutôt que de fautes.

cad) ou des réductions propres au langage SMS (ex : *slt* ou *tkt*). Suite à ces traitements préalables, tous les mots encore inconnus de nos lexiques sont considérés comme des mots à normaliser.

Dans un premier temps, nous montrerons comment nous avons extrait automatiquement nos règles de correction, puis nous expliquerons comment nous combinons ces règles entre elles. Enfin, nous verrons comment pondérer les candidats de correction/normalisation proposés par notre système afin de ne conserver que le ou les candidats les plus probables.

3.2 Apprentissage des règles de correction

L'apprentissage de nos règles de correction s'appuie sur un corpus de fautes lexicales annotées, le corpus WiCoPaCo (Max et Wisniewski, 2010) sur lequel nous reviendrons à la section 4.1. Nous verrons notamment comment nous en avons extrait un ensemble de couples de formes (forme fautive, forme bien orthographiée) correspondant à des fautes lexicales. Un tel ensemble constitue l'entrée de notre système, et nous cherchons à en extraire automatiquement des règles pondérées de correction. Comme indiqué précédemment, nous faisons l'hypothèse que chaque mot « fautif » est le résultat d'au plus une erreur « lourde ». Ainsi, pour générer nos règles de correction, il nous suffit d'extraire automatiquement de chaque couple de forme : le contexte gauche complet précédant la faute, la faute et sa correction, le contexte droit complet suivant la faute. Les contextes gauche et droit complets étant identiques dans la forme fautive et dans la forme bien orthographiée, il est simple d'extraire les séquences de lettres constituants une faute et représentant la correction de cette faute. Pour le couple *souevnt*::*souvent* par exemple, l'alignement se fera comme suit :

sou | ev | nt
↕ ↕ ↕
sou | ve | nt

Nous apprendrons ainsi les informations suivantes. :
– la *zone fautive* et sa contrepartie corrigée, c'est-à-dire les deux chaînes de caractères à substituer pour passer d'une forme à une autre : « $ev \rightarrow ve$ » ;
– le contexte dans lequel s'applique cette substitution : « *sou_nt* ».

Si nous conservons ces informations en l'état nous obtiendrons des règles trop spécifiques (de type : *{sou}{ev→ve}{nt}*) qui s'appliqueraient à trop peu de fautes. Afin d'éviter les cas de sous-correction et de sur-correction, nous avons construit deux jeux de règles en généralisant de deux façons différentes les règles individuelles directement extraites des fautes, ou *règles de base*. Pour définir ces deux jeux de règles, nous avons besoin de définir quelques termes. Étant donnée une zone fautive, nous appellerons dans la suite *contexte de niveau 1* le contexte formé des deux lettres encadrant immédiatement cette zone (la lettre à sa gauche et celle à sa droite). Nous appellerons *contexte de niveau 2* les deux lettres situées à une distance de 2 de la faute, c'est-à-dire les deux lettres encadrant immédiatement le contexte de niveau 1. Dès lors qu'un contexte va au-delà des limites du mot, nous utilisons la pseudo-lettre #.

Règles spécifiques Le premier jeu de règles est produit par généralisation limitée à partir des règles de base. Elles ne conservent que le type (consonne (C), voyelle (V) ou #) des lettres de contexte de niveau 2 mais laisse inchangé le contexte de niveau 1. Ainsi, si l'on rencontre la paire *souevnt*::*souvent* lors de l'apprentissage, la règle spécifique qui sera extraite est : *{Vu}{ev→ve}{nC}*.

Lorsqu'une nouvelle règle spécifique ne diffère d'une autre que par son contexte de niveau 1, nous les fusionnons : leurs contextes droits et gauches sont unifiés. Ceci constitue une généralisation par rapport aux deux règles prises isolément. Par exemple, si la faute suivante à traiter est *pievrt*::*pivert*, on obtient la règle de base *{Vi}{ev→ve}{rC}* : la règle de substitution extraite et le contexte de niveau 2 sont identiques à la règle précédente. Seul le contexte droit de niveau 1 diffère. Nous fusionnons alors les contextes de niveau 1 par disjonction, la règle obtenue étant alors notée : *{V[iu]}{ev→ve}{[nr]C}*. Cette fusion constitue une généralisation puisque cette règle couvre alors également les règles de base *{Vu}{ev→ve}{rC}* et *{Vi}{ev→ve}{nC}*, pourtant non attestées. En revanche, dès lors que le contexte de niveau 2 ne correspond pas, nous créerons une nouvelle règle. Ainsi, la faute *réserev*::*réserve* induira l'ajout d'une nouvelle règle à notre jeu de règle : *{Vr}{ev→ve}{##}*.

Règles larges Le second jeu de règles est produit par généralisation plus forte des règles de base : le contexte de niveau 2 est effacé, et seul le type des lettres du contexte de niveau est conservé. Pour la faute *souevnt*::*souvent*, la règle large induite sera donc : *{V}{ev→ve}{C}*.

Nous avons également appliqué certaines généralisations à ces deux jeux de règles, au niveau de la règle de substitution elle-même. Tout d'abord, nous avons généralisé les erreurs de duplication d'une lettre (ex : « *fautte* ») et de suppression d'une lettre doublée (ex : « *ereur* »), en les représentant comme suit :

fautte::*faute* : *{Vt}{+_→_}{e#}*
ereur::*erreur* : *{Vr}{_→+_}{eV}*

Si le symbole « + » est à droite du tiret bas, alors on duplique le contexte droit de niveau 1, s'il est à gauche, on duplique le contexte gauche de niveau 1. Selon que le « + » est à gauche ou à droite de la flèche, on couvre ainsi respectivement les cas de duplication manquante et de duplication erronnée. Nous avons par ailleurs généralisé les fautes d'accents et de cédille. Pour ce faire, nous représentons une lettre accentuée par la combinaison de deux caracères : l'accent en question suivi de la lettre concernée. Ainsi, nous noterons le mot « *arrêt* » comme ceci : « *ˆe* » et, pour la faute « arret », nous obtiendrons la règle suivante : *{Cr}{_→ˆ}{eC}*, qui couvre le rajout d'un accent circonflexe sur toute voyelle, pour peu que les contextes correspondent.

Ainsi, pour construire ces deux jeux de règles, nous parcourons chaque couple (forme mal orthographiée, forme corrigée) du corpus annoté et nous extrayons les règles spécifiques et larges comme décrit ci-dessus. Toutes ces règles sont pondérées comme suit. À chaque fois qu'une règle est extraite ou modifiée par un couple, son nombre d'« occurrences » est incrémenté de 1. Nous prenons alors le logarithme de ce nombre d'occurrences, puis nous le normalisons par transformation affine entre 0 (règle de nombres d'occurrence minimal) à 1 (règle de nombres d'occurence maximal). Le résultat constitue le poids de la règle. La table 1 illustre quelques exemples de règles de correction apprises par notre système.

TYPE DE LA RÈGLE	RÈGLE	POIDS	EXEMPLE DE FAUTES CONCERNÉES
Spécifique	{V[pfnlmctbsredg]}{_ → +_}{[aieuonyr]C}	0,970	**aten**dre → **atten**dre
	{V[rmltdvcsnpyxg]}{a → e}{[nmilu]C}	0,660	r**alant**ir → r**alent**ir
Large	{C}{io → oi}{C}	0,298	**tiol**e → **toil**e
	{[CV#]}{´ → ` }{V}	0,738	é**lé**ve → é**lè**ve

TABLE 1 – Exemples de règles de correction extraites

3.3 Génération de règles de correction génériques

Afin de pallier l'absence possible de règles de correction propre aux fautes typographiques nous avons généré un nouveau jeu de règle de correction. Par fautes typographiques nous entendons ici les fautes telles que l'ajout et la suppression d'une lettre (ex : *boneur*, *bonheure*), la substitution d'une lettre par une autre (ex : *bomheur*) et l'inversion de deux lettres ensemble (ex : *bohneur*). Ces quatre opérations sont à la base de la notion de distance d'édition (Damerau, 1964). Nous avons donc ajouter une seconde passe de correction, qui cherche tous les mots du L*efff* qui sont à une distance d'édition de 1 du mot « fautif ». Nous dénoterons ce module par le terme de *correction générique*.

Cette méthode, plus risquée que la précédente, est suceptible de provoquer du bruit dans nos résultats. C'est pourquoi la correction générique ne corrigera que les mots « fautifs » que notre système par règles n'aura pas traité. Chaque candidat proposé par le module de correction générique est pondéré par le logarithme de sa fréquence dans la Wikipedia française. Il est ensuite normalisé de façon affine de telle sorte qu'un mot inconnu de la Wikipédia, qui sera considéré comme y étant attesté 0,1 fois, obtienne un poids de 0 et que le mot le plus fréquent de la Wikipedia ait un poids de 1. Ainsi les mots les plus fréquents de la langue auront plus de chance d'être choisis comme candidat de correction lors de l'évaluation, dès lors que l'on prendra la fréquence en compte.

3.4 Génération et sélection des candidats de correction

Notre système vise à proposer les candidats de correction les plus probables pour une faute donnée, en espérant obtenir une ou plusieurs normalisations valides. Pour ce faire nous nous appuyons sur nos deux premiers ensembles de règles (spécifiques et larges). Chaque règle qui peut s'appliquer sur le mot « fautif » permet de générer un candidat de correction/normalisation, qui n'est retenu que s'il est présent dans notre lexique de référence, le L*efff*. En l'absence de candidat, nous autorisions chaque règle à être complétée par des changements d'accentuation. Pour cela, nous vérifions pour chaque candidat obtenu par l'application d'une règle si sa contrepartie non accentuée est présente dans une version préalablement désaccentuée du L*efff*. Si c'est le cas, nous considérons toutes les versions accentuées correspondantes comme des corrections possibles. Pour le mot *elevve* par exemple, nous proposerons *élevé* ou *élève* [3]. Dans le cas où nous n'avons toujours pas de candidat, nous tentons d'en produire à l'aide de nos règles de correction génériques.

3. Le nombre de fautes d'accentuation autorisé est le nombre minimum qu'il faut effectuer pour obtenir un candidat. Par exemple, pour le mot *elevvé*, seul *élevé* sera proposé car *élève* supposerait 3 modifications d'accentuation, alors qu'*élevé* n'en implique qu'une seule.

Nos deux premiers jeux de règles proposent chacuns des candidats de correction accompagnés de leurs scores de correction (cf. table 2) [4] pour un mot « fautif ». Les règles spécifiques proposeront automatiquement moins de candidats que les règles larges [5]. Par ailleurs, les scores proposés par ces deux jeux pour une même correction ne seront jamais identiques. Il faut donc pouvoir prendre en compte leur deux analyses afin d'attribuer à chaque candidat de correction un score global.

FAUTE	CANDIDATS PROPOSÉS	POIDS DES RÈGLES SPÉCIFIQUES	POIDS DES RÈGLES LARGES
onférence	conférence	0.459	0.754
	inférence	0.236	0.543
fitrage	titrage	_	0.188
	filtrage	_	0.781
	vitrage	0.001	0.094
arquéologues	archéologues	0.101	0.149
innacompli	inaccompli	0.351	0.094

TABLE 2 – Exemples de candidats (cas de base : sans changement d'accent, règles génériques non déclenchées)

Nous avons par ailleurs constaté que certaines propositions de correction correspondent à des mots assez rares dans la langue et sont, malgré tout, trop bien pondérés. Pour pallier cela, nous prenons en compte la fréquence de chaque candidat de correction dans la langue. Pour ce faire nous avons extrait un score de fréquence compris entre 0 et 1 de la même façon que pour la correction générique (cf. plus haut). Ainsi, pour évaluer, scorer et trier chaque candidat de correction c du mot d'origine w, nous prenons en compte trois paramètres :

1. le score $S_s(c, w)$ égal au poids de la règle spécifique qui permet de passer de w à c, s'il y en a une,
2. le score $S_l(c, w)$ égal au poids de la règle large qui permet de passer de w à c,
3. la score de fréquence $F(c)$ de la correction/normalisation proposée c.

Le score global d'un candidat sera ensuite calculé par combinaison linéaire de ces trois scores. Soit λ_s le coefficient assigné au score de la correction spécifique et λ_l celui assigné au score de la correction large. Nous définissons le score global d'une correction par $S_{\lambda_s,\lambda_l}(c, w) = \lambda_s S_s(c, w) + \lambda_l S_l(c, w) + (1 - \lambda_s - \lambda_l)F(c)$, où λ_s et λ_l sont compris entre 0 et 1. C'est à partir de ce score que nous définissons le meilleur ou les n meilleurs candidats pour une faute donnée. Naturellement, ce score dépend des valeurs choisies pour λ_s et λ_l. Nous avons donc fait varier ces deux paramètres lors de l'évaluation du scorage et donc du choix des candidats parmi l'ensemble des candidats proposés pour chaque faute. Au préalable, nous avons évalué la pertinence des candidats eux-mêmes.

4 Évaluation

4.1 Données d'entraînement et de test

Afin d'évaluer notre système, nous avons utilisé le corpus d'erreurs WiCoPaCo (Max et Wisniewski, 2010). Ce corpus, créé à partir des révisions des pages de la Wikipédia francophone, est composé de 72 483 erreurs lexicales et de 74 100 erreurs grammaticales qui ont été annotées comme telles dans le corpus par leurs auteurs au moyen d'un processus automatique. Chacune des fautes est associée à sa correction, effectuée par un contributeur de la Wikipédia. Puisque dans ce travail nous nous intéressons uniquement aux fautes lexicales, nous n'avons utilisé que les fautes lexicales, c'est-à-dire les fautes annotées « *non_word_error* ». Par ailleurs, nous ne voulons pas que la fréquence d'un mot puisse biaiser la pondération des règles qui en seront extraites. Nous n'avons donc conservé qu'une seule occurrence de chaque faute annotée (soit un total de 36 344 fautes). Nous avons utilisé 60% de ces dernières, soit 21 581 fautes, comme données d'entraînement pour l'apprentissage de nos deux jeux de règles [6]. Nous avons obtenu de la sorte un jeu de 4 795 règles

4. Si le candidat de correction résulte d'une correction par règle associé à une faute d'accentuation, son score de correction sera celui de la correction par règle. La faute d'accentuation ne modifie donc pas le poids de correction d'une règle.

5. Toute correction proposée par les règles spécifiques le sera par les règles larges. L'inverse n'est pas vrai.

6. Suite à plusieurs tests, nous avons constaté que prendre un corpus plus grand (90% de l'ensemble des fautes lexicales, par exemple) n'améliorait pas nos résultats de manière significative. Nous avons par ailleurs l'intention d'étendre nos tests en travaillant sur un autre corpus d'erreurs non décrit ici, en cours de développement, d'environ 20 000 fautes. Afin que nos expériences puissent être comparables, il est préférable que ces deux corpus fassent la même taille, c'est pourquoi nous n'avons conservé que 60% des fautes lexicales de WiCoPaCo pour l'apprentissage.

spécifiques et un jeu de 3 073 règles larges. Par ailleurs, environ 7,5% des fautes lexicales de WiCoPaCo (soit 2 731 fautes), sans recouvrement avec les données d'entraînement, ont été conservées pour constituer notre corpus de test.

Comme indiqué plus haut, notre système s'insère dans une chaîne complète de normalisation textuelle. Cette chaîne a notamment pour tâche, préalablement à l'application de notre système de correction par analogie, de détecter et de préserver de toute correction ultérieure différents types de mots inconnus, tels que les créations lexicales ou les mots étrangers. Toutefois, dans la mesure où nous cherchons ici à évaluer notre système de normalisation, indépendamment du reste d'une telle chaîne, nous avons d'une part désactivé les modules de détection des emprunts et des néologismes et d'autre part vérifié qu'il n'y avait pas de créations lexicales ou de mots étrangers dans nos données d'évaluation, supposées ne contenir que des fautes lexicales et leur correction. En réalité, nous en avons trouvé quelques cas ne relevant pas de fautes lexicales, mais plutôt de substitutions d'un mot par un autre. Nous avons ainsi identifié et retiré manuellement de nos données d'évaluation 38 occurrences de créations lexicales (ex : *herbologiste*, *windobe*, *zitanoisme*) ou à des mots étrangers (ex : *poesía*, *musgos*), qui ont été remplacées dans la Wikipedia par d'autres mots. Ces cas constituaient environ 1,4% des données d'évaluation, ce retrait n'a donc pas eu d'impact significatif sur nos résultats. Par ailleurs, 137 « fautes » ont été écartées parce qu'elles appartenaient déjà à notre lexique de référence, le Le*fff*, et ne constituaient pas à ce titre des candidats à la correction pour notre système (ex : *téléscope*, *soeur*). Suite à ces retraits, nos données de test contiennent 2 556 fautes lexicales pour lesquelles nous disposons d'une correction spécifiée par les contributeurs de la Wikipédia.

4.2 Résultats obtenus

Bien que nous ayons développé des techniques de correction automatique dans notre système, rappelons que notre objectif reste la normalisation textuelle. Nous avons donc évalué notre système aussi bien sur ses résultats en correction automatique qu'en normalisation. Ces deux évaluations se distinguent par leur manière de valider un candidat pour un mot « fautif » donné. En normalisation, on attendra de notre système qu'il nous propose n'importe quelle forme fléchie du lemme « correct ». En correction automatique, on n'acceptera que la forme normativement correcte du mot mal orthographié. Il se peut que, parmi les normalisations proposées pour une faute donnée, plusieurs soient valides alors qu'aucune ne corresponde pour autant à la bonne correction [7].

Nous avons évalué nos résultats en deux temps : (i) Nous nous sommes tout d'abord intéressés à la qualité des candidats proposés par notre système ; (ii) Puis nous nous sommes concentrés sur la pertinence de notre système de scorage. Pour la suite de cette évaluation, nous avons utilisé les notions habituelles de précision, rappel et f-mesure. Si l'on ne prend en compte qu'un seul candidat par faute, la précision P correspond au rapport entre le nombre de mots bien corrigés (resp. normalisés) et le nombre de mots pour lesquels nous avons proposé au moins un candidat de correction. Le rappel R sera calculé comme le rapport entre le nombre de mots bien corrigés (resp. normalisés) et le nombre de mots donnés en entrée. Si l'on cherche à prendre en compte plusieurs candidats, on considérera qu'un mot est bien corrigé/normalisé dès lors qu'une bonne correction/normalisation se trouve parmi lesdits candidats. La f-mesure F, quant à elle, sera calculée comme habituellement : $F = 2PR/(P + R)$.

4.2.1 Génération des candidats

Cette première évaluation, effectuée grâce aux données d'évaluation décrites ci-dessus, ne s'appuie sur aucun des scores proposés par notre système. Son but est principalement d'évaluer la qualité des candidats de correction/normalisation proposées par ce dernier. Elle permet de vérifier dans quelle mesure ces candidats ne contiennent pas trop de bruit, tout en proposant une correction/normalisation valide parmi les candidats proposés pour un mot « fautif » donné.

Nombre de candidats Nous avons commencé par vérifier que notre système ne proposait pas trop de candidats, notamment au niveau du module de correction générique. Nous avons donc compté les candidats obtenus avec et sans ce dernier. Le nombre de candidats proposés par notre système reste raisonnable, même avec le module de correction générique. En effet, comme le montre la figure 2, seule une faute sur quatre se voit attribuer plus de 4 candidats. Le nombre de ces derniers décroît par ailleurs assez rapidement puisque même avec le module de correction générique, près de 45% des mots « fautifs » de notre corpus de test ne se voient attribuer qu'une seule proposition de correction.

Couverture Cette figure nous montre aussi l'utilité de la correction générique. En se limitant aux seuls jeux de règles, seuls 5,3% des mots ne reçoivent aucun candidat. La correction générique permet de réduire ce taux à 3,6%. Les 93 mots

7. Pour la faute *dormion* par exemple, si notre système nous propose les candidats *dormir* ou *dormons*, nous considérerons que faute a été normalisée correctement bien que la correction attendue ne figure pas parmi les candidats.

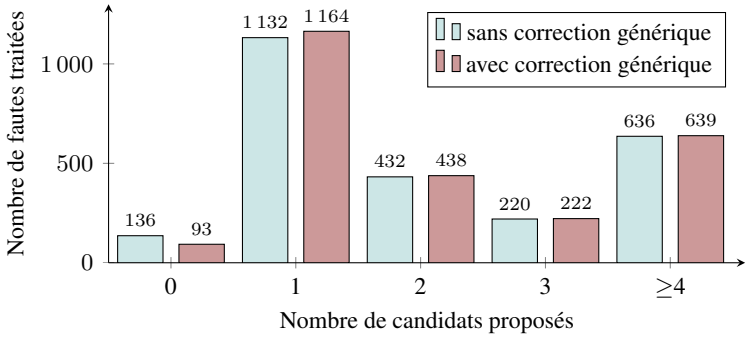


FIGURE 2 – Nombre de candidats par fautes avec et sans la correction générique

non traités ont été étudiés manuellement : 85% d'entre eux correspondent à des fautes trop complexes ou trop nombreuses (ex : *arondisemernts*, *aérdorme*) et 15% à des séquences de caractères difficilement interprétables (ex : *klàkoes*, *piwut*).

Par ailleurs, nous avons cherché à estimer la qualité des candidats proposés. Pour ce faire, étant donné un inventaire de candidats pour chaque faute, nous pouvons calculer une borne inférieure et une borne supérieure pour le système dans son ensemble. Ces bornes encadrent les performances que nous pourrons obtenir lorsque nous attribuerons des valeurs aux coefficients λ_r et λ_l, c'est-à-dire lorsque nous évaluerons la qualité de notre système de scorage à la section 4.2.2. Ces bornes ne prennent pas en compte les scores de correction de chaque candidat, et sont définies comme suit.

Borne supérieure pour le système complet Afin de déterminer le score maximum que notre système pourrait obtenir étant donné les candidats proposés, nous utilisons un oracle. Pour chaque faute, dès lors qu'une bonne correction est proposée, l'oracle le choisit. À défaut, si au moins une bonne normalisation est proposée, l'une d'entre elles est choisie au hasard par l'oracle. Si aucune normalisation valable ne se trouve parmi les candidats proposées par notre système, le choix de l'oracle importera peu. Les résultats obtenus avec un tel oracle, présentés dans le tableau 3, sont nécessairement meilleurs que ceux que nous obtiendrons avec notre système de scorage.

	Sans correction générique (éval sur 2 556 mots)		Avec correction générique (éval sur 2 556 mots)		Uniquement sur les fautes concernées par la correction générique (éval sur 93 mots)	
	correction	normalisation	correction	normalisation	correction	normalisation
précision	94,1	95,6	93,7	95,1	69,8	69,8
rappel	89,1	90,4	90,3	91,6	22,1	22,1
f-mesure	91,6	92,9	92	93,3	33,5	33,5

TABLE 3 – Évaluation des candidats de correction faite par l'oracle

Nous avons calculé la précision, le rappel et la f-mesure que notre système obtient avec et sans la correction générique afin de voir l'impact de cette dernière. Cette évaluation a été faite de manière systématique pour la tâche de correction automatique (en vérifiant que le mot attendu fait partie les candidats proposés) et pour la tâche de normalisation (en vérifiant qu'au moins un des candidats appartienne au même lemme que le token attendu). Bien que l'écart entre nos scores avec et sans la correction générique ne soit pas très élevé, cette dernière nous permet de traiter plus de mots (cf. ci-dessus), sans trop faire diminuer notre précision. Dans les deux dernières colonnes du tableau 3, nous pouvons par ailleurs faire l'observation suivante : bien que le rappel de la correction générique soit très faible (pour les raisons citées en section 4.2.1), sa précision reste acceptable (70% environ). Nous l'avons donc conservée dans la suite de nos expériences.

Par ailleurs, nous constatons que les scores obtenus pour les tâches de correction et de normalisation diffèrent peu. Cela montre que notre système de correction se trompe rarement dans la flexion du mot qu'il tente de corriger, dès lors qu'il a correctement identifié le bon lemme. On note que les scores du module de correction générique sont identiques en correction et en normalisation. Cela s'explique par le fait que la correction générique, n'effectuant que des opérations non pondérées sur un caractère, a peu de chances de proposer une correction fautive qui soit une forme fléchie du bon lemme [8].

8. Supposons ainsi que l'on demande au module de correction générique de proposer un candidat pour le mot fautif *prèt*. Il pourra proposer par exemple, grâce à une unique substitution, *prêt*, *prit*, *près*. Cet exemple illustre le fait que, de façon générale, les mots les plus « proches » (au sens de la distance de Levenshtein) d'une même faute ne sont pas tous, loin de là, des formes fléchies d'un même lemme. C'est d'autant plus vrai que le module de correction générique n'est appliqué qu'aux fautes suffisamment « complexes » ou « inattendues » pour que nos règles de correction, qui s'appliquent à plus de 96% des fautes, ne puissent pas s'appliquer.

Baseline pour le système complet Notre système de scorage doit être plus performant qu'un système de sélection aléatoire parmi les candidats proposés. C'est pour cette raison que nous avons choisi d'évaluer notre système en le laissant sélectionner aléatoirement l'un des candidats de correction associé à chaque faute. Les scores obtenus (cf. table 4) sont assez élevés pour un système effectuant un choix aléatoire. Cela montre que les propositions de correction/normalisation faites par notre système sont dans l'ensemble assez pertinentes.

	Sans correction générique (éval sur 2 556 mots)		Avec correction générique (éval sur 2 556 mots)		Uniquement sur les fautes sur lesquelles la correction générique est essayée (éval sur 93 mots)	
	correction	normalisation	correction	normalisation	correction	normalisation
précision	58,3	69,4	58,4	69,2	60,5	60,5
rappel	55,2	65,7	56,2	66,7	19,1	19,1
f-mesure	56,7	67,5	57,3	67,9	29,1	29,1

TABLE 4 – Évaluation des candidats de correction faite de manière aléatoire

4.2.2 Sélection des meilleurs candidats de normalisation

La correction générique améliorant nos résultats, nous avons choisi de la conserver dans notre système. Tous les scores figurants dans la suite de cette section proviendront donc de notre système de normalisation combinant nos jeux de règles de corrections et la correction générique.

Dans un premier temps, nous avons comparé la qualité de système sur les tâches de correction et de normalisation lorsque l'on ne conserve pour chaque faute que le candidat de score maximal [9]. Nous avons ensuite évalué notre système lorsque l'on conserve les deux puis les trois meilleurs candidats, configuration qui est pertinente si l'on décide par exemple d'en ressortir à un modèle de langage ou à tout autre module ultérieur pour effectuer le choix final, ou bien si l'étape de normalisation est un préalable à un traitement capable de prendre en entrée un graphe de mots [10]. Pour l'ensemble de ces configurations, nous avons procédé à de multiples évaluations en faisant varier les poids assignés aux jeux de règles et à la fréquence, c'est à dire en faisant varier la valeur de λ_r et de λ_l (la fréquence étant pondérée par $1 - \lambda_r - \lambda_l$) [11].

La somme des valeurs de λ_r et de λ_l doit être égale ou inférieure à 1. Nous avons donc commencé par tester toutes les combinaisons possibles de ces deux coefficients en les faisant varier entre 0 et 1 avec un pas de 0,1. Toutefois, il est apparu que ces coefficients étaient trop élevés et empêchaient la prise en compte de la fréquence. En effet, même si les scores de fréquence sont normalisés entre 0 et 1 (tout comme ceux de nos jeux de règles), ces derniers restent très faibles sur la grande majorité des candidats. Nous avons donc réévalué notre système en faisant cette fois-ci varier λ_r et λ_l entre 0 et 0,001 avec un pas de 0,0001. Nous illustrons ces derniers résultats aux figures 3 à 8 sur la précision et la f-mesure. En théorie, les résultats de notre système peuvent varier entre les bornes inférieures et supérieures calculées à la section 4.2.1, à savoir respectivement 69,2% et 95,1% pour la précision et respectivement 67,9% et 93,3% pour la f-mesure.

Les figures 3 et 4 illustrent la précision et la f-mesure obtenues par notre système lorsque l'on ne conserve que le candidat de score maximum pour chaque faute. Plus la zone d'une figure est claire, plus la précision ou la f-mesure représentée dans cette figure sera élevée. Par ailleurs, nous avons inséré des courbes de niveau (iso-précision ou iso-f-mesure) : on peut voir par exemple sur ces deux figures que les meilleurs scores pour la précision s'élèvent à plus de 87,2 (la précision maximum atteinte est en réalité de 87,7%) et ceux concernant la f-mesure à plus de 85,7 (la f-mesure maximale atteinte est en réalité de 86,1%). Ces scores sont donc bien supérieurs aux bornes inférieures calculées précédemment. La similarité entre les figures 3 et 4 s'explique par le fait que les scores de rappel obtenus, bien que plus faibles que ceux de la précision, se répartissent de la même manière en fonction des coefficients λ_r et λ_l. À titre indicatif, lorsque λ_r et λ_l varient entre 0 et 0,001, le rappel est presque partout entre 83% et 84%, avec un minimum à 81,4% (lorsque $\lambda_r = \lambda_l = 0$) et un maximum

9. Lorsque plusieurs candidats ont le même score et qu'il est maximal, l'un d'entre eux est choisi aléatoirement.
10. Des expériences conservant les cinq candidats les mieux classés ont donné des résultats très proches de ceux obtenus avec les trois meilleurs.
11. Plutôt que chercher la valeur optimale de nos trois variables de cette façon, nous aurions pu nous appuyer sur un système reposant sur la maximisation de l'entropie ou sur un perceptron et ainsi les apprendre automatiquement. Néanmoins une telle approche ne répondrait pas parfaitement à l'objectif que nous nous sommes posé ici. Nous ne voulons en effet pas classifier les corrections proposées en deux catégories mais les ordonner selon la confiance que l'on a en le fait qu'ils soient des corrections/normalisations valides de la faute à traiter. Nous avons toutefois tenté d'évaluer notre système avec des scores obtenus avec un système par maximum d'entropie. Les résultats obtenus furent néanmoins beaucoup moins bons que ceux présentés précédemment (par exemple, sur la meilleure normalisation, nous perdons près de 5.5% de précision et de F-mesure si le système prend en compte le biais, 7% sinon).

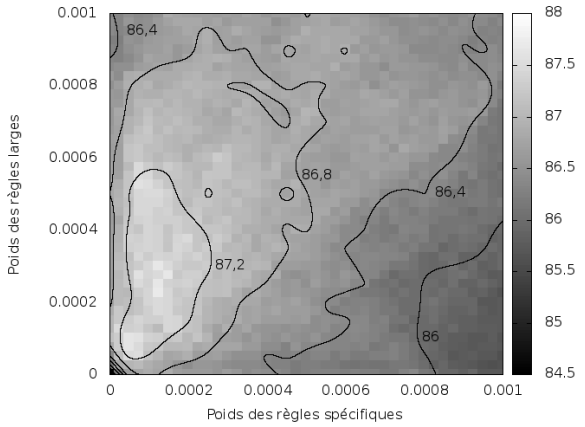


FIGURE 3 – Précision pour la meilleure normalisation

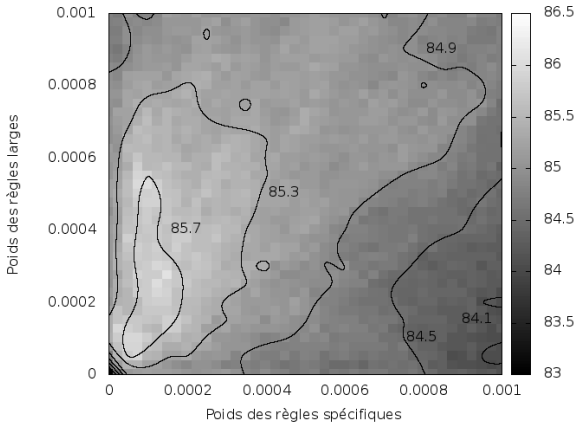


FIGURE 4 – F-mesure pour la meilleure normalisation

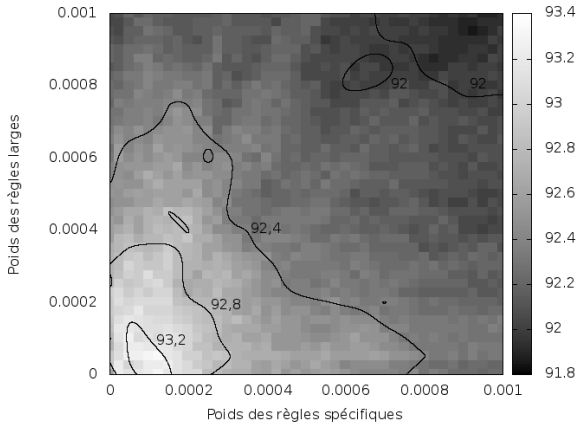


FIGURE 5 – Précision pour les 2 meilleures normalisations

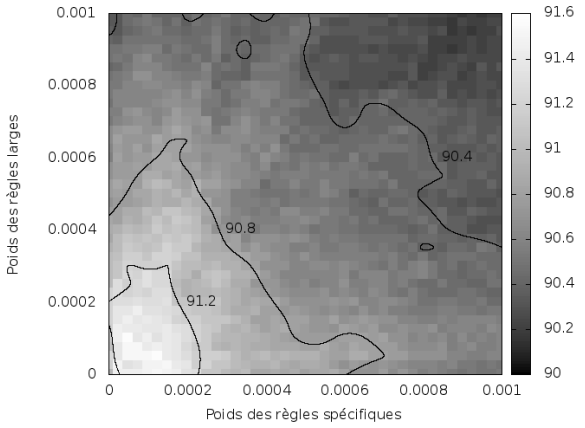


FIGURE 6 – F-mesure pour les 2 meilleures normalisations

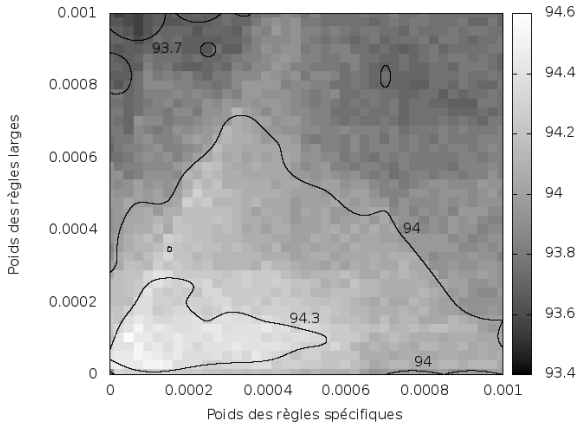


FIGURE 7 – Précision pour les 3 meilleures normalisations

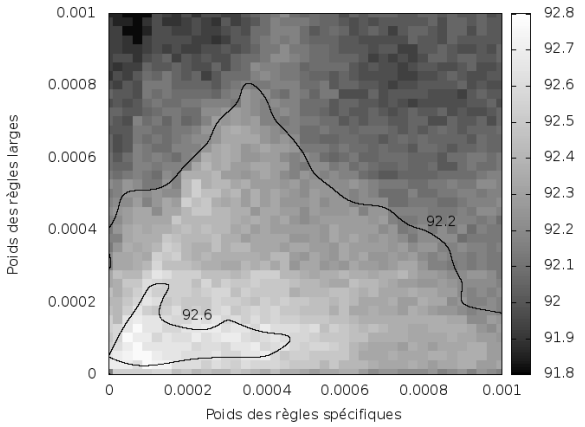


FIGURE 8 – F-mesure pour les 3 meilleures normalisations

à 84,3%. Ces deux figures démontrent clairement l'utilité des scores proposés par nos règles lorsque l'on ne garde que le meilleur candidat : si nous ne prenons que la fréquence en compte ($\lambda_r = \lambda_l = 0$), notre f-mesure est de 82,9%, soit 3,2% de moins que la f-mesure maximale). Bien que de façon moins nette, ce constat reste vrai lorsque l'on conserve les deux ou trois meilleurs candidats les mieux scorés. La f-mesure gagne ainsi 1,5% si on prend en compte les règles pour deux candidats de correction et 0,7% pour trois candidats.

Bien que le passage d'un candidat à deux nous permette d'améliorer nos scores de près de 5% absolus, le passage de deux candidats à trois est légèrement moins significatif. Cela s'explique notamment par le faible nombre de corrections

proposées pour chaque mot « fautif ». Notre système n'attribue trois candidats ou plus qu'à environ 40% de notre corpus de test (cf. figure 2). Le nombre de mots « fautifs » pour lesquels nous devons sélectionner les meilleurs candidats est par conséquent automatiquement réduit. Autre conséquence de ce fait : nos scores de précision et de f-mesure se rapprochent de la borne supérieure théorique dès la prise en compte de seulement deux candidats. En effet, notre précision s'élève à plus de 93,3% (cf. figure 5) et la f-mesure à plus de 91,5% (cf. figure 6).

Ces figures montrent que nous obtenons nos meilleurs résultats lorsque l'on donne plus de poids aux règles larges qu'aux règles restreintes. En effet, bien que les règles restreintes soit plus précises et plus fiables, elles s'appliquent moins souvent. Les règles larges peuvent ainsi détecter et corriger plus de fautes. Ces deux jeux sont donc complémentaires : nous obtenons de meilleurs scores en associant les deux. Enfin, la prise en compte de la fréquence d'un mot pour sa normalisation reste primordiale. Cela est particulièrement visible dans les figures 5, 6, 7 et 8 dans lesquelles nos scores se dégradent dès lors que λ_r ou λ_l sont au-dessus de 0,0006 environ, et ce d'autant plus que λ_r ou λ_l sont élevés.

5 Conclusion

La présence de fautes ou de formes non standard peuvent mettre à mal l'analyse d'un texte si cette analyse n'est pas adaptée aux textes bruités. Par exemple, pour un outil d'extraction d'informations, il sera plus difficile de détecter les motifs intéressants si le texte à traiter est dégradé. Le travail présenté ici vise à participer à l'amélioration des performances d'un tel outil en proposant une étape intermédiaire de prétraitement, dont l'objectif est de normaliser, au moins partiellement, des données textuelles bruitées. Pour ce faire, nous avons mis en place et évalué un système de normalisation fonctionnant à l'aide de règles de correction induites par analogie. L'objectif de ce système est de proposer un ou plusieurs candidats de normalisation pondérés pour tous les mots inconnus de nos lexiques présents dans un texte.

Les résultats obtenus étant satisfaisants, nous voudrions tout d'abord intégrer ce premier système à notre chaîne de traitement complète, qui inclut notamment des étapes de détection des néologismes et des emprunts, afin d'évaluer ses performances et son apport de manière plus globale. Par ailleurs, nous nous sommes pour l'instant limités aux fautes lexicales. À terme, nous aimerions étendre ce travail aux fautes grammaticales en prenant en compte le contexte dans lequel apparaissent ces fautes, par exemple au moyen de modèles de langage. D'autre part, ce système n'a pour l'instant été évalué que sur le français. L'apprentissage de nos règles de correction étant faite automatiquement, cette normalisation pourrait parfaitement fonctionner pour d'autres langues, pour peu qu'il existe une base de fautes corrigées pour cette langue. Cette base pourrait parfaitement, par exemple, être extraite de la Wikipedia correspondante, de la même façon qu'a été construit le corpus WiCoPaCo pour le français ou comme le propose Zesch (2012) pour l'anglais. Enfin, l'apprentissage de nos règles de correction est actuellement effectuée à partir d'un corpus de fautes annotées. Il n'existe actuellement que très peu de corpus de ce type. C'est pourquoi nous aimerions à terme pouvoir apprendre nos règles de façon non-supervisée, à partir d'un corpus brut légèrement bruité. Cela nous permettrait d'obtenir un système qui ne dépendrait d'aucune ressource particulière mis à part d'un échantillon de la langue et d'un lexique de cette langue.

Références

BEAUFORT, R., ROEKHAUT, S., COUGNON, L.-A. et FAIRON, C. (2010). A Hybrid Rule/Model-Based Finite-State Framework for Normalizing SMS Messages. *In Proceedings of ACL'10*, pages 770–779, Uppsala, Suède.

BLAIR, C. R. (1960). A program for correcting spelling errors. *Information and Control*, 3(1):60–67.

BOYD, A. (2009). Pronunciation modeling in spelling correction for writers of English as a foreign language. *In Proceedings of the HLT-NAACL'09 Student Research Workshop and Doctoral Consortium*, pages 31–36, Boulder, Colorado.

BRILL, E. et MOORE, R. C. (2000). An Improved Error Model for Noisy Channel Spelling Correction. *In Proceedings of ACL'00*, Hong Kong.

CARLSON, A. et FETTE, I. (2007). Memory-based context-sensitive spelling correction at web scale. *In Proceedings of ICMLA'07*, pages 166–171, Cincinnati, Ohio.

DAMERAU, F. (1964). A technique for computer detection and correction of spelling errors. *Comm. ACM*, 7(3):171–176.

GUIMIER DE NEEF, E., DEBEURME, A. et PARK, J. (2007). TiLT correcteur de SMS : évaluation et bilan qualitatif. *In Actes de TALN'07*, pages 123–132, Toulouse, France.

HAN, B. et BALDWIN, T. (2011). Lexical normalisation of short text messages : makn sens a #twitter. *In Proceedings of JACL-HTL'11*, pages 368–378, Portland, États-Unis.

HATHOUT, N. (2010). Morphonette : a morphological network of French. *Proceedings of CoRR*, abs/1005.3902.

KERNIGHAN, M. D., CHURCH, K. W. et GALE, W. A. (1990). A Spelling Correction Program Based on a Noisy Channel Model. *In Proceedings of CoLing'90*, pages 205–210, Helsinki, Finland.

KOBUS, C., YVON, F. et DAMNATI, G. (2008). Transcrire les SMS comme on reconnaît la parole. *In Actes de TALN'08*, pages 128–138, Avignon, France.

KUKICH, K. (1992). Techniques for Automatically Correcting Words in Text. *ACM Computing Surveys*, 24(4):377–439.

LAVALLÉE, J.-F. et LANGLAIS, P. (2011). Moranapho : un système multilingue d'analyse morphologique fondé sur l'analogie formelle. *TAL*, 52(2):17–44.

LAVALLÉE, J. F. et LANGLAIS, P. (2009). Unsupervised morphological analysis by formal analogy. *In Multilingual Information Access Evaluation Vol. I : Text Retrieval Experiments, Proceedings of CLEF*, pages 618–625, Corfu, Greece.

LEPAGE, Y. (1998). Solving analogies on words : An algorithm. *In Proceedings of CoLing-ACL'98*, pages 728–735, Montreal, Quebec, Canada.

LEPAGE, Y. (2000). Languages of analogical strings. *In Proceedings of CoLing*, pages 488–494, Saarbrücken, Germany.

LEPAGE, Y. et DENOUAL, E. (2005). Purest ever example-based machine translation : Detailed presentation and assessment. *Machine Translation*, 19(3-4):251–282.

LEVENSHTEIN, V. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10(8):707.

LI, M., ZHANG, Y., ZHU, M. et ZHOU, M. (2006). Exploring distributional similarity based models for query spelling correction. *In Proceedings of ACL-CoLing'06*, pages 1025–1032, Syndey, Australie.

MAX, A. et WISNIEWSKI, G. (2010). Mining naturally-occurring corrections and paraphrases from wikipedia's revision history. *In Proceedings of LREC'10*, Valletta, Malta.

MITTON, R. (1996). *English Spelling and the computer*. London :Longman.

MITTON, R. (2010). Fifty years of spellchecking. *Writting Systems Research*, 2(1):1–7.

MOREAU, F., CLAVEAU, V. et SÉBILLOT, P. (2007). Automatic morphological query expansion using analogy-based machine learning. *In Proceedings of ECIR'07*, Rome, Italie.

OFLAZER, K. (1996). Error-tolerant finite-state recognition with applications to morphological analysis and spelling correction. *Computational Linguistics*, 22(1):73–89.

PARK, Y. A. et LEVY, R. (2011). Automated whole sentence grammar correction using a noisy channel model. *In Proceedings of ACL-HTL'11*, pages 934–944, Portland, Oregon, USA.

SAGOT, B. (2010). The L*efff*, a freely available and large-coverage morphological and syntactic lexicon for French. *In Proceedings of LREC'10*, La Valette, Malta.

SAGOT, B. et BOULLIER, P. (2008). SxPipe 2 : architecture pour le traitement pré-syntaxique de corpus bruts. *TAL*, 49(2):155–188.

SAGOT, B., NOUVEL, D., MOUILLERON, V. et BARANES, M. (2013). Extension dynamique de lexiques morphologiques pour le français à partir d'un flux textuel. *In Actes de TALN'13*, Les Sables d'Olonne, France.

SEDDAH, D., SAGOT, B., CANDITO, M., MOUILLERON, V. et COMBET, V. (2012). The French Social Media Bank : a Treebank of Noisy User Generated Content. *In Proceedings of CoLing'12*, Mumbai, Inde.

STROPPA, N. et YVON, F. (2005). An analogical learner for morphological analysis. *In Proceedings of CoNLL'05*, pages 120–127, Stroudsburg, PA, USA.

STROPPA, N. et YVON, F. (2006). Du quatrième de proportion comme principe inductif : une proposition et son application à l'apprentissage de la morphologie. *TAL*, 47(1):33–59.

SUIGNARD, P. et KERROUA, S. (2013). Utilisation de contextes pour la correction automatique ou semi-automatique de réclamations clients. *In Actes de TALN'13*, Les Sables d'Olonne, France.

TOUTANOVA, K. et MOORE, R. C. (2002). Pronunciation Modeling for Improved Spelling Correction. *In Proceedings of ACL'02*, pages 144–151, Philadelphie, États-Unis.

VÉRONIS, J. (1988). Computerized correction of phonographic errors. *Computers and the Humanities*, 22(1):43–56.

YVON, F. (2011). `spellChecker` : un système de correction automatique fondé sur des automates probabilistes. Livrable du Projet TRACE (ANR-09-CORD-023). Accessible à l'URL http ://anrtrace.limsi.fr/dev/Anr_trace_-_lot3.pdf.

ZESCH, T. (2012). Detecting malapropisms using measures of contextual fitness. *TAL*, 53(3):11–31.

Une évaluation approfondie de différentes méthodes de compositionalité sémantique

Antoine Bride Tim Van de Cruys Nicolas Asher
IRIT, Université Paul Sabatier, 118 route de Narbonne, F-31062 TOULOUSE CEDEX 9
[nom]@irit.fr

Résumé. Au cours des deux dernières décennies, de nombreux algorithmes ont été développés pour capturer la sémantique des mots simples en regardant leur répartition dans un grand corpus, et en comparant ces distributions dans un modèle d'espace vectoriel. En revanche, il n'est pas trivial de combiner les objets algébriques de la sémantique distributionnelle pour arriver à une dérivation d'un contenu pour des expressions complexes, composées de plusieurs mots. Notre contribution a deux buts. Le premier est d'établir une large base de comparaison pour les méthodes de composition pour le cas adjectif_nom. Cette base nous permet d'évaluer en profondeur la performance des différentes méthodes de composition. Notre second but est la proposition d'une nouvelle méthode de composition, qui est une généralisation de la méthode de Baroni & Zamparelli (2010). La performance de notre nouvelle méthode est également évaluée sur notre nouveau ensemble de test.

Abstract. In the course of the last two decades, numerous algorithms have sprouted up that successfully capture the semantics of single words by looking at their distribution in text, and comparing these distributions in a vector space model. However, it is not straightforward to construct meaning representations beyond the level of individual words – i.e. the combination of words into larger units – using distributional methods. Our contribution is twofold. First of all, we carry out a large scale evaluation, comparing different composition methods within the distributional framework for the case of adjective-noun composition, making use of a newly developed dataset. Secondly, we propose a novel method for adjective-noun composition, which is a generalization of the approach by Baroni & Zamparelli (2010). The performance of our novel method is equally evaluated on our new dataset.

Mots-clés : sémantique lexicale, sémantique distributionnelle, compositionalité.

Keywords: lexical semantics, distributional semantics, compositionality.

1 Introduction

Au cours des deux dernières décennies, il y a eu un intérêt croissant dans les méthodes dites « distributionnelles » pour la sémantique lexicale (Landauer & Dumais, 1997; Lin, 1998; Turney & Pantel, 2010). Ces méthodes sont nommées ainsi car elles se fondent sur l'hypothèse distributionnelle (Harris, 1954), qui stipule que les mots qui apparaissent dans les mêmes contextes ont tendance à être sémantiquement similaire. Dans l'esprit de cet adage, maintenant bien connu, de nombreux algorithmes ont été développés pour tenter de capturer la sémantique des mots simples en regardant leur répartition dans un grand corpus, et en comparant ces distributions dans un modèle d'espace vectoriel.

En comparaison avec les études manuelles de la sémantique formelle lexicale, cette approche apporte une couverture bien plus vaste et une analyse d'une grande masse de données empiriques. En revanche, il n'est pas trivial de combiner les objets algébriques de la sémantique distributionnelle pour arriver à une dérivation d'un contenu pour des expressions complexes, composées de plusieurs mots. *A contrario*, l'opération de l'application et des représentations qui utilisent le formalisme du λ-calcul dans la sémantique formelle nous donne des méthodes de composition générales et sophistiquées qui peuvent traiter non seulement la composition de sens dans les cas simples mais aussi des phénomènes complexes comme la coercition ou la composition avec des formules finement typées (Asher, 2011; Luo, 2010; Bassac *et al.*, 2010). Malgré des efforts pour trouver une méthode générale de composition et diverses approches proposées pour la composition des structures syntaxiques spécifiques (par exemple adjectifs et syntagmes nominaux, ou verbes transitifs et objets (Mitchell & Lapata, 2008; Coecke *et al.*, 2010; Baroni & Zamparelli, 2010)), le problème de composition demeure un défi pour

l'approche distributionnelle. De plus, la validation des méthodes de composition proposées s'est souvent faite à petite échelle (Mitchell & Lapata, 2008). Bien que ces études sur les jugements de similarité soient prometteuses et significatives, il serait intéressant d'avoir des études ayant une plus large couverture de validation. Elles nous permettraient de mieux comparer les différentes méthodes de composition proposées.

Notre contribution a deux buts. Le premier est d'établir une large base de comparaison pour les méthodes de composition pour le cas adjectif_nom. Pour cela nous avons créé un vaste ensemble de test utilisant des paires contenant une expression composée (adjectif_nom) et un nom qui doit être proche sinon identique sémantiquement de l'expression composée. Ces paires ont été extraites semi-automatiquement du Wiktionnaire français. Cette base de paires similaires nous permet d'évaluer en profondeur la performance des différentes méthodes de composition. Nous avons testé trois méthodes de composition déjà existantes, à savoir l'approche additive et multiplicative (Mitchell & Lapata, 2008), ainsi que l'approche par fonctions lexicales (Baroni & Zamparelli, 2010).

Les deux premières méthodes sont complètement générales et s'appliquent à des vecteurs que l'on peut automatiquement calculer pour les adjectifs et noms. En revanche, l'approche de Baroni et Zamparelli nécessite d'apprendre une fonction particulière associée à chaque adjectif. Notre second but est de généraliser l'approche fonctionnelle afin d'éliminer le besoin de conserver une fonction par adjectif. Pour cela nous utilisons une fonction généralisée apprise à l'aide des fonctions d'adjectifs de l'approche de Baroni et Zamparelli. Cette fonction généralisée se combine alors avec le vecteur d'un adjectif et celui d'un nom de manière entièrement générale. La performance de notre nouvelle méthode de l'approche fonctionnelle généralisée est également évaluée sur notre ensemble de test.

Nous avons organisé notre contribution de façon suivante. Nous détaillons d'abord les différents modèles de composition que nous évaluons dans notre étude, avec un rappel sur les différentes méthodes de composition existantes et puis une description de notre généralisation de l'approche fonctionnelle. Puis nous décrivons notre méthode d'évaluation et les résultats. Après une section sur les travaux connexes aux nôtres, nous concluons et nous précisons quelques pistes de travaux futurs.

2 Modèles de composition

Nous expliquons, dans cette section, quels modèles de composition ont été testés et à quoi ceux-ci correspondent. Après une bref rappel des modèles de composition simples, nous détaillons notamment la méthode des fonctions lexicales de Baroni & Zamparelli (2010) ainsi que la généralisation que nous en avons faite.

Voici les notations utilisées dans la suite. Lorsque nous décrivons un objet théorique, sans soucis de sa représentation physique par l'ordinateur, nous utilisons la police de base. Quand nous discutons de vecteurs, ceux-ci sont écrits en **gras**. Les matrices sont représentées en **MAJUSCULES GRASSES**. Enfin, nous écrivons les tenseurs [1] d'ordre 3 avec une majuscule calligraphiée, par exemple $\mathscr{A}$. De plus, comme nous ne manipulons pas de tenseur d'ordre supérieur à 4, nous appelons simplement les tenseurs d'ordre 3 « tenseurs » [2]. Pour conclure, le coefficient d'indice i d'un vecteur $\mathbf{v}$ est noté $\mathbf{v}_i$; la notation des coefficients des matrices et tenseurs se fait de manière analogue.

Dans la suite de cet article les adjectifs seront représentés par la lettre « a » et les noms par la lettre « n ».

2.1 Modèles de composition simples

Trois modèles de composition que nous avons utilisés sont simples à décrire : les méthodes triviale, additive et multiplicative. L'approche triviale, notée C_t et que nous utilisons comme base de comparaison, ignore l'adjectif :

$$C_t(a,n) = \mathbf{n}$$

Le modèle additif, noté C_a, consiste à réaliser une combinaison linéaire entre les vecteurs $\mathbf{a}$ et $\mathbf{n}$ à l'aide de coefficients indépendants de ceux-ci :

$$C_a(a,n) = \alpha\,\mathbf{n} + \beta\,\mathbf{a}$$

1. Un tenseur est la généralisation d'une matrice à plusieurs indices. Pour une introduction sur les tenseurs, regardez Kolda & Bader (2009).
2. les tenseurs d'ordre 1 étant les vecteurs et les tenseurs d'ordre 2 les matrices.

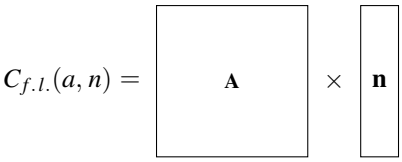


FIGURE 1: Composition dans l'approche par fonctions lexicales

Enfin, tandis que les deux rois faisaient chanter des *Te Deum*, chacun dans son camp, [Candide] prit le parti d'aller raisonner ailleurs des effets et des causes. Il passa par-dessus des tas de morts et de mourants, et gagna d'abord un **village** voisin ; il était en cendres : c'était un **village** abare que les Bulgares avaient brûlé, selon les lois du droit public. [...]
Candide s'enfuit au plus vite dans un autre **village** : il appartenait à des Bulgares, et les héros abares l'avaient traité de même...

FIGURE 2: extrait de *Candide* de Voltaire, Chapitre 3

Sur un ensemble de développement, nous avons testé le modèle pour différentes valeurs de α et β telles que $\alpha+\beta=1$ [3] et conservé les valeurs donnant les meilleurs résultats : $\alpha=0.4$ et $\beta=0.6$.

Le modèle multiplicatif, noté C_m, consiste à multiplier les vecteurs **a** et **n** terme à terme :

$$C_m(a,n) = \mathbf{n} \otimes \mathbf{a}$$
$$\text{où} \quad (\mathbf{n} \otimes \mathbf{a})_i = \mathbf{n}_i \times \mathbf{a}_i$$

L'approche par fonctions lexicales de Baroni & Zamparelli (2010) étant plus complexe, nous la décrivons dans la section suivante. Nous expliquons ensuite pourquoi et comment nous avons tenté de généraliser cette approche.

2.2 Fonctions Lexicales

Le modèle de composition par fonctions lexicales, noté $C_{f.l.}$, consiste à représenter les adjectifs par des matrices. Ainsi la combinaison d'un adjectif et d'un nom est le produit de la matrice **A** et du vecteur **n** comme le montre la figure 1.

L'approche distributionelle ne permet cependant pas de générer naturellement des matrices. Baroni et Zamparelli proposent donc d'apprendre la matrice d'un adjectif à partir d'exemples de vecteurs nom_adjectif obtenus directement à partir du corpus. De tels vecteurs nom_adjectif sont obtenus de la même manières que des vecteurs représentant un seul mot : quand la combinaison de l'adjectif et du nom occurre, on observe son contexte. Prenons l'exemple du paragraphe en figure 2. Le mot « village » apparaît trois fois. La première occurrence peut contribuer à créer le vecteur **village_voisin**, la deuxième à créer **village_abare**, et la dernière à créer **village_autre**.

Une fois que l'on a créé suffisamment de vecteurs nom_adjectif pour un adjectif donné, on calcule la matrice **A**. Pour cela, on réalise une régression partielle des moindres carrés, sur les combinaisons nom_adjectif. Formellement, en notant $\mathbf{n_a}$ les vecteurs nom_adjectif, il s'agit de trouver **A** minimisant :

$$\sum_{\mathbf{n}} \| \mathbf{A} \times \mathbf{n} - \mathbf{n_a} \|_2 \qquad \text{où} \quad \| \mathbf{v} \|_2 = \sqrt{\sum_i \mathbf{v}_i^2}$$

Pour reprendre l'exemple précédent, on minimiserait, notamment, $\|\mathbf{VOISIN} \times \mathbf{village} - \mathbf{village\_voisin}\|_2$ pour obtenir la matrice **VOISIN**.

Il est important de noter qu'une telle approche nécessite un corpus plus important que les autres approches. En effet, comme il ne s'agit plus seulement d'observer des exemples d'utilisation d'adjectifs ou de noms isolés mais des exemples d'utilisation de la combinaison d'un adjectif et d'un nom, les occurrences sont intrinsèquement plus rares. Dans le paragraphe en figure 2, chacune des apparitions du mot « village » peut contribuer à la création du vecteur **village** mais aucune ne peut contribuer à la création du vecteur **village_félon**.

3. Les vecteurs étant normalisés (*cf.* 3.2), cette condition ne réduit pas la généralité de notre test.

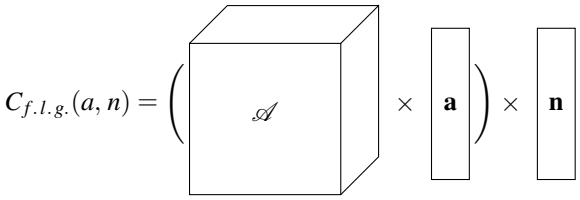


FIGURE 3: Composition dans l'approche par fonction lexicale généralisée

Baroni & Zamparelli (2010) expliquent comment limiter les problèmes liés au manque d'exemples. De plus, les expériences présentées jusqu'à maintenant montrent que les corpus actuels permettent une implémentation efficace de l'approche par fonctions lexicales pour les adjectifs les plus courants. En effet, celle-ci a obtenu les meilleurs résultats sur un certain nombre d'expériences.

Néanmoins, l'approche de Baroni et Zamparelli reste limitée pour traiter des adjectifs relativement rares. Par exemple, l'adjectif « félon » apparaît 217 fois dans le corpus FRWaC (Baroni *et al.*, 2009). C'est assez pour générer un vecteur **félon**, mais très peu pour espérer générer un nombre suffisant de vecteurs **nom_félon** et donc générer la matrice **FÉLON**.

De plus, devoir apprendre une matrice pour chaque adjectif pose un problème théorique. En effet, cette approche suppose, comme l'approche de Montague, que l'effet d'un adjectif sur un nom est idiosyncratique à l'adjectif (Kamp, 1975). Mais le désavantage de ceci est que les données montrent que la plupart des adjectifs dans les langues du monde sont subjectifs et se comportent selon des principes générales de composition (Partee, 2010; Asher, 2011). La manière dont les adjectifs sont utilisés dans la langue française laisse supposer qu'il existe une façon générale de combiner adjectifs et noms. Lorsque l'on connaît la signification d'un adjectif, l'association à un nom est rarement problématique. Ceci, indépendamment de la présence ou de l'absence d'exemples d'association.

2.3 Généralisation

Pour résoudre ces problèmes, nous proposons de généraliser les fonctions lexicales que sont les matrices d'adjectifs par une fonction lexicale unique : le tenseur de composition adjectivale $\mathscr{A}$. Dans notre approche, notée $C_{f.l.g.}$, la combinaison d'un adjectif et d'un nom est le produit du tenseur $\mathscr{A}$ avec le vecteur adjectif puis le vecteur nom, *c.f.* figure 3.

On peut noter que le produit du tenseur $\mathscr{A}$ et du vecteur **a** est une matrice dépendante de l'adjectif et multipliée au vecteur **n**. Cette matrice correspond à la matrice **A** de l'approche par fonctions lexicales de Baroni et Zamparelli. Ainsi, comme le montre la figure 4, nous obtenons le tenseur $\mathscr{A}$ à l'aide d'exemples de matrices obtenues par la méthode de Baroni et Zamparelli, et de vecteurs obtenus naturellement dans l'approche distributionelle. Plus précisément nous effectuons une régression partielle des moindres carrés sur les matrices générées par les équations. Formellement, il s'agit de trouver $\mathscr{A}$ minimisant :

$$\sum_a \| \mathscr{A} \times \mathbf{a} - \mathbf{A} \|_2 \qquad \text{où } \| \mathbf{M} \|_2 = \sqrt{\sum_{i,j} \mathbf{M}_{i,j}^2}$$

Cette équation ressemble beaucoup à l'équation non généralisée. En effet, dans les deux cas, l'objectif formel est de trouver une application linéaire [4] minimisant des équations dans un espace vectoriel de dimension finie.

Cependant, la généralisation fait une hypothèse bien plus forte. En effet, l'image d'une application linéaire est toujours de dimension inférieure à son espace de départ. Or, par construction, si les vecteurs **adjectif** existent dans un espace de dimension N, alors les matrices **ADJECTIF** existent dans un espace de dimension $N \times N$. Ainsi, s'il existe un tenseur $\mathscr{A}$, cela signifie que le sous-espace engendré par les matrices **ADJECTIF** est de taille inférieur à N réduisant considérablement leur degré de liberté maximal initial (dimension $N \times N$). L'approche de Baroni et Zamparelli n'a pas cette hypothèse puisque les vecteurs **nom** et **nom_adjectif** coexistent dans un même espace (et les espaces engendrés par ceux-ci ont donc la même dimension maximale).

Moins formellement, chercher une telle application linéaire présuppose qu'il n'y pas plus d'« information » dans le sous-espace d'arrivée que de le sous-espace de départ. Cela créé une différence notable entre les deux méthodes. En effet, dans la méthode de Baroni et Zamparelli, les vecteurs **nom** et **nom_adjectif** existent dans le même espace. L'hypothèse sus-cité

4. représenté par une matrice ou un tenseur.

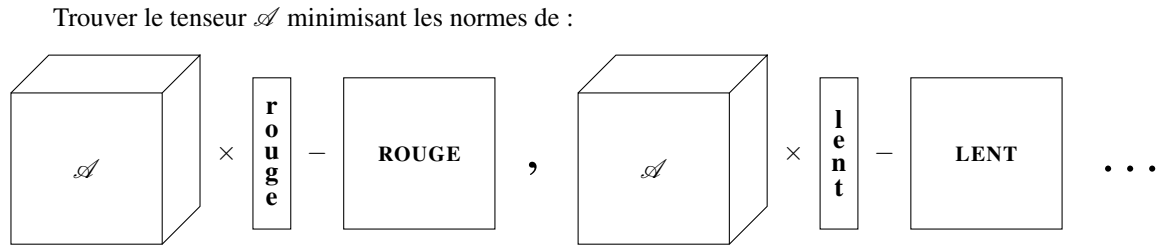


FIGURE 4: Apprentissage du tenseur $\mathscr{A}$

consiste donc à supposer que l'on a pas besoin de plus d'informations pour décrire **voisin_village** que pour décrire **village** ; uniquement d'informations différentes. Cette hypothèse est partagée par beaucoup de méthodes de composition dont l'objectif est de pouvoir réaliser des compositions en cascade [5]. *A contrario*, dans la méthode généralisée, l'application linéaire recherchée a un espace de départ [6] bien plus petit que son espace d'arrivée [7]. La généralisation fait donc une hypothèse beaucoup plus forte : elle suppose que les matrices **A** créées par la méthode de Baroni et Zamparelli ne sont pas plus informatives que les vecteurs **a** que l'on peut extraire directement d'un *corpus*. Ces matrices ne seraient, d'une certaine manière, qu'une réécriture des adjectifs pertinente pour la composition.

Ceci étant, de manière similaire à l'approche de Baroni et Zamparelli, notre approche nécessite d'apprendre un nombre significatif de matrices **A**. Cela n'est pas un problème, car le FRWaC fournit suffisamment d'adjectifs sur lesquels l'approche de Baroni et Zamparelli fonctionne parfaitement. Par exemple, le 2000$^{\text{ième}}$ adjectif le plus courant dans le FRWaC (« fasciste ») y occurre plus de 4000 fois.

Pour reprendre l'exemple de l'adjectif « félon », notre approche exige uniquement de connaître le vecteur **félon**, évitant le problème de manque de données lié à la construction de la matrice FÉLON.

Une fois le tenseur $\mathscr{A}$ obtenu, il nous fallait vérifier expérimentalement sa pertinence. En effet, nous n'avions pas garantie que le tenseur optimisant les équations décrites dans la figure 4 soit intéressant sémantiquement.

3 Évaluation

3.1 Description de la tâche

Pour évaluer les différents modèles de composition, nous avons construit une tâche de similarité inspirée des travaux de Zanzotto *et al.* (2010) et utilisée pour la tâche *evaluating phrasal semantics* de SEMEVAL-2013 (Korkontzelos *et al.*, 2013). La tâche propose de juger la similarité entre une combinaison adjectif_nom et un seul nom. Ceci est important, étant donné que les modèles de composition doivent être capables de traiter des combinaisons adjectifs_nom de taille arbitraire. La tâche est donc la suivante :

Soient **comb** = Combinaison(adjectif, nom1) et **nom2**
Évaluer Similarité(**comb**, **nom2**)

La « Combinaison » est réalisé par les différents modèles de composition. La « Similarité » doit être une fonction binaire ; les valeurs de retour étant « similaire » et « non_similaire ». Cependant, l'approche distributionnelle ne fournit naturellement que des valeurs de similarité continues (*e.g.* cosinus entre deux vecteurs). Nous avons donc pris des exemples positifs et des exemples négatifs de notre ensemble de test afin de savoir quelles valeurs de cosinus correspondent à « similaire » et quelles valeurs de cosinus correspondent à « non_similaire ». Plus précisément, nous avons, pour chaque approche, réalisé une régression logistique sur 50 exemples positifs et 50 exemples négatifs (dorénavant séparés de notre ensemble de test) afin d'apprendre le seuil de cosinus à partir duquel une paire est similaire.

Nous avons créé notre ensemble de test d'une manière semi-automatique, en utilisant des dictionnaires. Prenons par exemple la définition de *champagne* dans le Wiktionnaire français [8], figure 5. D'une telle définition, il est assez simple

5. Par exemple, obtenir le sens de « grosse voiture rouge » en composant « grosse » et « voiture » puis « rouge » et « grosse voiture ».
6. l'espace des vecteurs **mot**.
7. l'espace des matrices **ADJECTIF**.
8. `http://fr.wiktionary.org/wiki/champagne`, accédé à 20 février 2014.

d'extraire la paire (mousseux_vin, champagne). En traversant un grand dictionnaire, il est ainsi possible d'extraire des paires (`adjectif_nom`, `nom`) positives (similaires).

champagne /ʃɑ̃.paɲ/ masculin

1. Vin mousseux produit en Champagne et protégé par une appellation d'origine contrôlée.
2. *(Histoire des techniques)* Cercle de fer pour soutenir l'étoffe à teindre dans la cuve de teinture.

FIGURE 5: Définition de *champagne*, extrait du Wiktionnaire français

Nous avons donc téléchargé toutes les entrées du Wiktionnaire français, et nous les avons tagées avec le tageur MElt (Denis *et al.*, 2010). Ensuite, nous avons sélectionné les définitions qui débutent avec une combinaison adjectif-nom. Enfin, nous avons supprimé les instances utilisant des mots qui apparaissent trop peu fréquemment dans notre corpus FRWaC [9].

Les instances ainsi extraites d'une manière automatique étaient alors contrôlées manuellement. Toutes les paires jugées incorrectes étaient rejetées. Nous avons ainsi obtenu 714 exemples positifs.

Nous avons ensuite créé un premier fichier d'exemples négatifs en sélectionnant deux noms (`nom1`, `nom2`) et un adjectif `adjectif` aléatoirement. Les couples (`adjectif_nom1`, `nom2`) ainsi créés étaient ensuite vérifiés manuellement. Nous avons ainsi obtenu 899 exemples négatifs.

L'inconvénient d'un tel procédé est qu'il propose souvent des combinaisons `adjectif_nom1` insensées. Ceci simplifie la tâche de séparer exemples positifs et négatifs. Nous avons donc créé un second fichier d'exemples négatifs en sélectionnant des combinaisons `adjectif_nom1` depuis le Wiktionnaire et des noms `nom2` aléatoires. Nous avons ensuite vérifié manuellement que les couples (`adjectif_nom1`, `nom2`) ainsi créés était bien des exemples négatifs. Nous avons ainsi obtenu 494 exemples négatifs.

La table 1 montre 5 exemples positifs et 5 exemples négatifs de chaque sorte. Dans cette table, les noms et adjectifs sont sous forme de lemme. On peut noter que les exemples négatifs générés complètement aléatoirement contiennent des combinaisons `adjectif_nom` ayant un sens clair (penchant_autoritaire) et n'en n'ayant pas (chasse_fossile). Les exemples négatives créés à base du Wiktionnaire, en revanche, contiennent uniquement des combinaisons qui ont un sens clair. [10]

exemples positifs	exemples négatifs aléatoires	exemples négatifs Wiktionnaire
(mot_court, abréviation)	(importance_fortuit, gamme)	(jugement_favorable, discorde)
(ouvrage_littéraire, essai)	(penchant_autoritaire, ile)	(circonscription_administratif, fumier)
(compagnie_honorifique, ordre)	(auspice_aviaire, ponton)	(mention_honorable, renne)
(costume_féminin, ensemble)	(banquette_celeste, discipline)	(attitude_hautain, racine)
(partie_unitaire, élément)	(chasse_fossile, propulsion)	(examen_attentif, condamnation)

TABLE 1: Une partie des ensembles de test.

3.2 Espace sémantique

Une fois le test choisi et les fichiers de test réalisés nous avons créé l'espace sémantique. Pour cela, nous avons utilisé le corpus FRWaC (Baroni *et al.*, 2009) – un corpus de 1,6 milliard de mots extrait du web – tagé avec le tageur MElt (Denis *et al.*, 2010) et parsé à l'aide du parseur MaltParser (Nivre *et al.*, 2006), formé sur une version de dépendances du *French treebank* (Candito *et al.*, 2010). Nous avons d'abord récupéré les lemmes des mots, adjectifs, et noms du corpus. Nous avons uniquement conservé les lemmes écrits en toutes lettres [11] puis selectionné les 10000 lemmes les plus fréquents pour chaque catégorie (mots, adjectifs, noms). Enfin, nous avons généré l'espace en utilisant les adjectifs et les noms

9. *i.e.* moins de 200 fois pour les adjectifs et moins de 1500 fois pour les noms.
10. Nous fournissons les fichiers correspondant sur simple demande par e-mail.
11. Cette étape élimine principalement les dates, les nombres en chiffre, et la ponctuation. Nous estimons que ceux-ci ont un intérêt limité en approche distributionnelle.

comme vecteurs, et les mots comme dimensions en utilisant la méthode des *bags of words*. Nous avons alors nettoyé l'espace ainsi créés en normalisant les vecteurs et en appliquant le *positive point-wise mutual information* (*ppmi*, (Church & Hanks, 1990)) à l'espace.

Nous avons alors comparé les méthodes sur trois versions de l'espace : l'espace entier, l'espace réduit à 300 dimensions par la méthode de décomposition en valeurs singulières (*svd*, (Golub & Van Loan, 1996)), et l'espace réduit à 300 dimensions par la méthode de factorisation en matrices positives (*nmf*, (Lee & Seung, 2000)). Nous avons fait cela pour pouvoir tester chaque méthode dans des conditions optimales. En effet :

– Un espace non réduit contient plus d'informations. Ainsi les méthodes compatibles (additive et multiplicative) peuvent obtenir de meilleurs résultats. Cependant, utiliser la méthode des fonctions lexicales sur l'espace non réduit demanderait d'apprendre des matrices de taille 10000×10000. Ceci poserait des problèmes de temps de calcul et de parcimonie des données comme on a vu ci-dessus. De même pour les fonctions étendues.
– Un espace réduit avec la méthode *svd* permet expérimentalement d'obtenir de bon résultats pour les fonctions lexicales. Cependant, la présence de valeurs négatives dans le vecteurs de l'espace réduit drastiquement l'efficacité de l'approche multiplicative.
– Un espace réduit avec la méthode *nmf* ne pénalise pas les approches multiplicatives.

3.3 Résultats

Les espaces sémantiques ayant été créés, nous avons d'abord testé les différentes approches sur le jeu de test utilisant des exemples négatifs complètement aléatoires (deuxième colonne de la table 1). Nous présentons les résultats dans la table 2a. Plusieurs commentaires peuvent être faits. Nous commentons d'abord les méthodes individuellement, puis nous les comparons.

	triviale	multiplicative	additive	fonctions lexicales	f. l. généralisées
non-réduit	0.83	0.86	0.88	N/A	N/A
svd	0.79	0.55	0.84	**0.93**	0.61
nmf	0.78	0.83	0.79	0.90	0.88

(a) Les exemples négatifs sont créés entièrement aléatoirement.

	triviale	multiplicative	additive	fonctions lexicales	f. l. généralisées
non-réduit	0.78	0.79	**0.83**	N/A	N/A
svd	0.77	0.55	0.82	**0.84**	0.46
nmf	0.75	0.73	0.79	0.78	0.78

(b) Les exemples négatifs sont créés à l'aide de combinaisons adjectif-nom existantes.

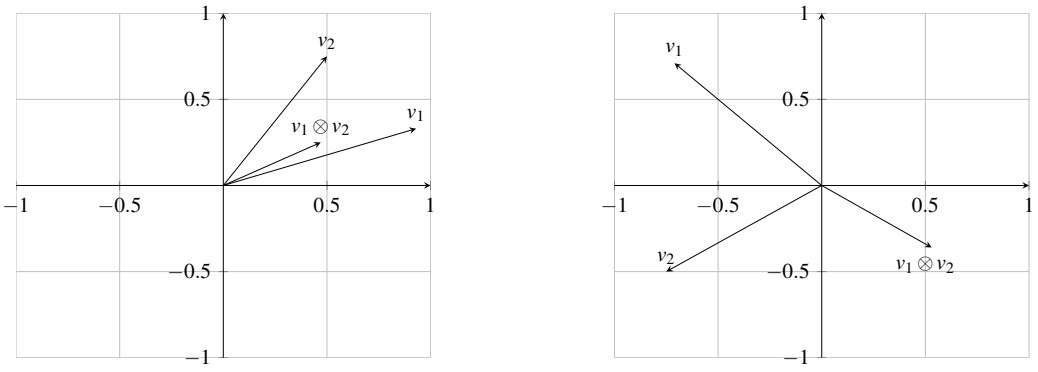
TABLE 2: Pourcentage de couples $(\text{adjectif\_nom1}, \text{nom2})$ bien classés selon l'approche et l'espace.

D'abord, l'approche triviale, consistant à comparer les deux noms et ignorer l'adjectif, affiche un taux de réussite relativement élevé ($\sim 80\%$). Ceci est dû au fait que la plupart des adjectifs ne changent pas la nature du nom auquel ils sont accolés. Une voiture rouge, lente, grosse, ou ancienne reste fondamentalement une voiture. Une voiture miniature n'est plus nécessairement une voiture mais de tels exemples sont rares.

Ensuite, la méthode multiplicative a de mauvaises performances sur l'espace réduit à l'aide de *svd*. Cela confirme l'incompatibilité de cette méthode avec les valeurs négatives générées par *svd*. La figure 6 permet de visualiser la raison à cela. On peut y voir que multiplier terme à terme deux vecteurs ayant des valeurs négatives résulte en un troisième vecteur très éloigné des deux autres. Cela va à l'encontre de l'idée selon laquelle la combinaison d'un nom et d'un adjectif à un sens proche du nom d'origine.

De plus, nous constatons que le modèle multiplicatif sur l'espace non-réduit n'atteint pas des résultats sensiblement meilleurs que le modèle trivial. La différence entre le modèle multiplicatif (0.86) et le modèle trivial (0.83) n'est pas statistiquement significative ($\chi^2 = 2.69$, $p > 0.05$). [12] Le modèle additif, en revanche, atteint un résultat en mode non-réduit (0.88) qui est significativement meilleur que la méthode triviale ($\chi^2 = 24.83$, $p < 0.01$) et le modèle multiplicatif

12. Pour tous nos tests de signification, nous utilisons le test de McNemar (Dietterich, 1998).



(a) vecteurs à valeurs positives

(b) vecteurs à valeurs négatives

FIGURE 6: l'effet de valeurs négatives sur l'approche multiplicative

($\chi^2 = 21.33$, $p < 0.01$). Les résultats du modèle additif pour les espaces *svd* et *nmf* sont également significatifs ($\chi^2 = 11.82$, $p < 0.01$ et $\chi^2 = 18.91$, $p < 0.01$, respectivement) mais il sont inférieurs au résultat de l'espace non-réduit. On constate que le modèle multiplicatif atteint un résultat de 0.83 dans l'espace *nmf* qui est significativement meilleur que le modèle multiplicatif ($\chi^2 = 31.34$, $p < 0.01$), mais toujours inférieur au résultat de l'espace non-réduit.

Ensuite, nous constatons que l'approche par fonctions lexicales de Baroni et Zamparelli dans l'espace *svd* obtient des résultats qui sont significativement meilleurs que toute autre approche avec tout autre espace ($\chi^2 = 33.49$, $p < 0.01$ pour la différence avec le modèle additif non-réduit). Nous constatons également que la fonction lexicale généralisée dans l'espace *nmf* obtient des résultats qui sont comparables avec l'approche de Baroni et Zamparelli dans ce même espace ($\chi^2 = 3.95$, $p > 0.01$) et équivalents aux meilleurs résultats des autres méthodes (notamment le modèle additif non-réduit). Cependant, la fonction lexicale généralisée – comme le modèle multiplicatif – a de faibles performances sur l'espace *svd* (0.61). Cela semble signifier que, dans cet espace, les matrices de la méthode des fonctions lexicales ne sont pas générables à l'aide d'un tenseur unique et des vecteurs correspondants.

Nous avons ensuite répété nos tests avec des exemples négatifs utilisant les combinaisons adjectif_nom extraites du Wiktionnaire français (troisième colonne de la table 1). Nous présentons les résultats dans la table 2b. Nous constatons que les résultats de nos premiers tests sont largement confirmés. Le modèle additif en espace non-réduit atteint un score qui est significativement meilleur que la méthode triviale (0.83 vs. 0.78, $\chi^2 = 10.69$, $p < 0.01$), bien que le modèle multiplicatif ne donne pas un score supérieur à la méthode triviale. Nous notons, toutefois, que l'approche par fonctions et l'approche additive obtiennent désormais des résultats globalement équivalents dans leurs meilleurs conditions respectives — 0.83 pour le modèle additif non-réduit vs. 0.84 pour le modèle fonctionel *svd*, une différence non-significative ($\chi^2 = 0.20$, $p > 0.05$). Cela semble indiquer que les fonctions lexicales sont particulièrement efficaces pour séparer les combinaisons insensées, mais qu'ils obtiennent un score inférieur quand ils doivent juger la similarité de compositions réelles.

4 Travaux connexes

Un certain nombre de chercheurs a déjà étudié et évalué divers modèles de composition au sein d'un cadre distributionnel. Une des premières tentatives pour évaluer de manière systématique des modèles simples de composition a été faite par Mitchell & Lapata (2008). Ils explorent un certain nombre de modèles différents pour la composition de vecteurs, dont les plus importants sont le modèle additif et le modèle multiplicatif. Ils évaluent leurs modèles sur une tâche de similitude de phrases nom-verbe. Pour évaluer leur modèle, ils ont demandé à des annotateurs humains de juger la similarité entre deux paires compositionnelles (par exemple en attribuant un certain score). La tâche du modèle de composition est alors de reproduire les jugements humains. Les résultats montrent que le modèle multiplicatif ainsi qu'une combinaison pondérée du modèle additif et du modèle multiplicatif donnent les meilleurs résultats. Les auteurs ont refait leur étude dans Mitchell & Lapata (2010) avec un ensemble de test plus large (les paires d'adjectifs et noms étaient également inclues), et ils ont confirmé leur résultats initiaux. Bien qu'une telle tâche de similitude a ses mérites, l'attribution d'un score de similitude est

plutôt difficile pour des juges humains [13]. Une décision binaire, comme dans notre tâche, est beaucoup moins floue. Nous soutenons que l'approche adoptée dans notre contribution donne une image plus claire et plus stable de la performance des différents modèles de composition.

Baroni & Zamparelli (2010) évaluent leur modèle de fonctions lexicales dans un contexte quelque peu différent. Ils évaluent leur modèle en regardant la capacité de reconstruire les vecteurs **nom_adjectif** qui n'ont pas été vus pendant la phase d'entraînement. Leur résultats montrent que leur modèle de fonctions lexicales atteint les meilleurs résultats pour reconstruire les vecteurs de co-occurrence originaux, suivi de près par le modèle additif. Notez que nous observons la même tendance dans notre évaluation.

Grefenstette *et al.* (2013) proposent aussi une généralisation du modèle de fonctions lexicales par des tenseurs. Leur généralisation a pour objectif différent, à savoir modéliser les verbes transitifs à l'aide de tenseurs. Cependant, nous utilisons un approche très similaire pour l'obtention des tenseurs. En effet, ils utilisent la méthode de Baroni et Zamparelli pour apprendre des matrices correspondant à une combinaison VERBE_COMPLÉMENT que l'on peut multiplier à un vecteur **sujet**, pour obtenir le vecteur **sujet_verbe_complément**. Par exemple MANGER_VIANDE multiplié au vecteur **chien** permet d'obtenir **chien_manger_viande**. Ils apprennent alors un tenseur correspondant à chaque verbe de la même manière que nous apprenons le tenseur $\mathscr{A}$.

Coecke *et al.* (2010) présentent un cadre théorique abstrait dans lequel un vecteur de phrase est une fonction du produit de Kronecker de ses vecteurs de mots, ce qui permet une plus grande interaction entre les différentes traits de mots. Un certain nombre d'instanciations du modèle de Coecke *et al.* (2010) – où l'idée clé est que les mots relationnels (par exemple les verbes) ont une structure riche (multidimensionnelle) qui agit comme un filtre sur leurs arguments – sont testés expérimentalement dans les articles de Grefenstette & Sadrzadeh (2011a) et Grefenstette & Sadrzadeh (2011b). Les auteurs évaluent leurs modèles en utilisant une tâche de similitude semblable à celle de Mitchell & Lapata. Cependant, ils utilisent des constructions compositionnelles plus étendues : plutôt que d'utiliser des compositions de deux mots (par exemple *verbe* et *objet*), ils utilisent des phrases simples transitives (*sujet verbe objet*). Ils montrent que leurs instanciations du modèle catégoriel obtiennent des meilleurs résultats que les modèles additifs et multiplicatifs sur leur tâche de similitude transitive.

Socher *et al.* (2012) présentent un modèle compositionnel basé sur les réseaux de neurones récursifs. Chaque nœud dans un arbre syntaxique est attribué à la fois un vecteur et une matrice ; le vecteur capture la signification réelle du constituant, tandis que la matrice modélise la manière dont il change le sens des mots et expressions voisins. L'évaluation s'est faite extrinsèquement, en utilisant le modèle dans une tâche de prédiction du sentiment. Ils montrent que l'approche basée sur les réseaux de neurones obtient de meilleurs résultats que les modèles additifs, multiplicatifs, et par fonctions lexicales. Cependant, d'autres chercheurs ont rapporté des résultats différents. Blacoe & Lapata (2012) évaluent les modèles additifs et multiplicatifs ainsi que l'approche de Socher *et al.* (2012) sur deux tâches différentes : la tâche de similitude de Mitchell & Lapata (2010) et une tâche de détection de paraphrases. Ils trouvent que les modèles additifs et multiplicatifs atteignent des meilleurs scores que le modèle de Socher *et al.* (2012).

Étroitement liée aux travaux sur la compositionnalité est la recherche sur le calcul de sens du mot en contexte. Erk & Padó (2008, 2009) font usage de restrictions sélectionnelles pour exprimer le sens d'un mot dans son contexte ; le sens d'un mot en présence d'un argument est calculé en multipliant le vecteur du mot avec un vecteur qui capture les restrictions sélectionnelles inverses de l'argument. Thater *et al.* (2009, 2010) étendent l'approche fondée sur les restrictions sélectionnelles en incorporant des co-occurrences du deuxième ordre dans leur modèle. Dinu & Lapata (2010) proposent un cadre probabiliste qui modélise la signification des mots comme une distribution de probabilité sur des facteurs latents. Cela permet de modéliser le sens contextualisé comme un changement dans la distribution du mot originel. Dinu et Lapata utilisent la factorisation de matrice positive (NMF) pour induire les facteurs latents.

En général, les modèles latents se sont avérées utiles pour la modélisation du sens des mots. L'un des modèles latents de la sémantique les plus connus est l'analyse de sémantique latente (LSA, Landauer & Dumais (1997)), qui utilise la décomposition en valeurs singulières afin d'induire automatiquement des facteurs latents de matrices terme-document. Un autre modèle de sens latent bien connu, qui adopte une approche générative, est l'allocation Dirichlet latente (LDA, Blei *et al.* (2003)).

Les tenseurs ont été utilisés auparavant pour la modélisation du langage naturel. Giesbrecht (2010) décrit un modèle de factorisation de tenseurs pour la construction d'un modèle distributionnel qui est sensible à l'ordre des mots. Et Van de Cruys (2010) utilise un modèle de factorisation de tenseurs afin de construire un modèle de restrictions sélectionnelles

13. En témoignent le faible taux d'accord d'inter-annotateur – Mitchell & Lapata (2010) rapportent une corrélation entre juges humains assez faible de 0.52 pour les combinaisons adjectif_nom.

multidimensionnelles de verbes, sujets et objets.

5 Conclusion

Dans notre contribution, nous avons testé différentes méthodes principales de compositionalité en approche distributionelle. À notre connaissance, nous sommes les premiers à réaliser de tels tests sur la langue française. Nous avons, de plus, créé un nouveau ensemble de test pour l'évaluation de la compositionalité dans un cadre distributionnel pour la langue française, librement disponible pour d'autres chercheurs.

Nos tests confirment que la méthode des fonctions lexicales de Baroni et Zamparelli a de bonnes performances en comparaison des autres approches. Nos tests semblent nuancer ceci par le fait que ces performances ne sont sensiblement meilleures que lorsque les exemples négatifs sont entièrement aléatoire.

De plus, nous avons proposé une généralisation de la méthode des fonctions lexicales. D'après nos tests, cette généralisation ne peut pas se faire dans les conditions optimales pour la méthode des fonctions lexicales. Ainsi bien que notre généralisation fonctionne correctement, les conditions dans lesquelles elle est utilisée font qu'elle a des résultats équivalent aux méthodes additive et multiplicative de Mitchell et Lapata, mais légèrement inférieurs à ceux de l'approche de Baroni et Zamparelli.

Dans le futur, il serait intéressant de tester différentes valeurs de réduction dimensionnelle afin d'optimiser notre fonction lexicale généralisée. De plus, il est possible que de meilleurs résultats puissent être obtenus en proposant plusieurs fonctions généralisées plutôt qu'une. On peut tenter, par exemple, de séparer les adjectifs intersectifs [14] des adjectifs non-intersectifs [15].

Il serait également intéressant de réaliser un ensemble de test pour une tâche avec laquelle la méthode des fonctions lexicales n'est pas entièrement compatible, comme la combinaison de noms. En effet, pour obtenir le sens de « laboratoire d'analyses médicales », il faut appliquer « analyses médicales » à « laboratoire ». Or la méthode des fonctions lexicales ne propose pas de manière satisfaisante d'obtenir la matrice ANALYSE_MÉDICALE. En effet, obtenir une telle matrice par apprentissage à partir d'exemples d'utilisation d'« analyse médicale » est en contradiction avec le principe de compositionalité.

Remerciements

Nous tenons à remercier toute l'équipe du projet composes [16] pour leur boite à outils DisSeCT (Dinu *et al.* (2013)) qui nous a sûrement épargné plusieurs mois de développement.

Références

ASHER N. (2011). *Lexical Meaning in Context : A Web of Words*. Cambridge University Press.

BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, **43**(3), 209–226.

BARONI M. & ZAMPARELLI R. (2010). Nouns are vectors, adjectives are matrices : Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, p. 1183–1193, Cambridge, MA : Association for Computational Linguistics.

BASSAC C., MERY B. & RETORÉ C. (2010). Towards a Type-theoretical account of lexical semantics. *Journal of Logic, Language and Information*, **19**(2), 229–245.

BLACOE W. & LAPATA M. (2012). A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 546–556, Jeju Island, Korea : Association for Computational Linguistics.

14. « Rouge » par exemple. Une voiture rouge est une voiture.
15. « Faux » par exemple. Une fausse voiture n'est pas une voiture.
16. `http://clic.cimec.unitn.it/composes/`

BLEI D. M., NG A. Y. & JORDAN M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, **3**, 993–1022.

CANDITO M., CRABBÉ B., DENIS P. *et al.* (2010). Statistical french dependency parsing : treebank conversion and first results. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*, p. 1840–1847.

CHURCH K. W. & HANKS P. (1990). Word association norms, mutual information & lexicography. *Computational Linguistics*, **16**(1), 22–29.

COECKE B., SADRZADEH M. & CLARK S. (2010). Mathematical foundations for a compositional distributed model of meaning. *Lambek Festschrift, Linguistic Analysis, vol. 36*, **36**.

DENIS P., SAGOT B. *et al.* (2010). Exploitation d'une ressource lexicale pour la construction d'un étiqueteur morpho-syntaxique état-de-l'art du français. In *Traitement Automatique des Langues Naturelles : TALN 2010*.

DIETTERICH T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, **10**(7), 1895–1923.

DINU G. & LAPATA M. (2010). Measuring distributional similarity in context. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, p. 1162–1172, Cambridge, MA.

DINU G., PHAM N. & M. B. (2013). Dissect : Distributional semantics composition toolkit. In *Proceedings of the System Demonstrations of ACL*, p. 31–36, East Stroudsburg PA : Association for Computational Linguistics.

ERK K. & PADÓ S. (2008). A structured vector space model for word meaning in context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, p. 897–906, Waikiki, Hawaii, USA.

ERK K. & PADÓ S. (2009). Paraphrase assessment in structured vector space : Exploring parameters and datasets. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, p. 57–65, Athens, Greece.

GIESBRECHT E. (2010). Towards a matrix-based distributional model of meaning. In *Proceedings of the NAACL HLT 2010 Student Research Workshop*, p. 23–28 : Association for Computational Linguistics.

GOLUB G. H. & VAN LOAN C. F. (1996). *Matrix Computations (3rd Ed.)*. Baltimore, MD, USA : Johns Hopkins University Press.

GREFENSTETTE E., DINU G., ZHANG Y.-Z., SADRZADEH M. & M. B. (2013). Multi-step regression learning for compositional distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS)*, p. 131–142, East Stroudsburg PA : Association for Computational Linguistics.

GREFENSTETTE E. & SADRZADEH M. (2011a). Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, p. 1394–1404, Edinburgh, Scotland, UK. : Association for Computational Linguistics.

GREFENSTETTE E. & SADRZADEH M. (2011b). Experimenting with transitive verbs in a discocat. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, p. 62–66, Edinburgh, UK : Association for Computational Linguistics.

HARRIS Z. S. (1954). Distributional structure. *Word*, **10**(23), 146–162.

KAMP H. (1975). Two theories about adjectives. *Formal semantics of natural language*, p. 123–155.

KOLDA T. G. & BADER B. W. (2009). Tensor decompositions and applications. *SIAM Review*, **51**(3), 455–500.

KORKONTZELOS I., ZESCH T., ZANZOTTO F. M. & BIEMANN C. (2013). Semeval-2013 task 5 : Evaluating phrasal semantics. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2 : Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, p. 39–47, Atlanta, Georgia, USA : Association for Computational Linguistics.

LANDAUER T. & DUMAIS S. (1997). A solution to Plato's problem : The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychology Review*, **104**, 211–240.

LEE D. D. & SEUNG H. S. (2000). Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13*, p. 556–562.

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 (COLING-ACL98), Volume 2*, p. 768–774, Montreal, Quebec, Canada.

LUO Z. (2010). Type-theoretical semantics with coercive subtyping. *SALT20, Vancouver*.

MITCHELL J. & LAPATA M. (2008). Vector-based models of semantic composition. *proceedings of ACL-08 : HLT*, p. 236–244.

MITCHELL J. & LAPATA M. (2010). Composition in distributional models of semantics. *Cognitive Science*, **34**(8), 1388–1429.

NIVRE J., HALL J. & NILSSON J. (2006). Maltparser : A data-driven parser-generator for dependency parsing. In *Proceedings of LREC-2006*, p. 2216–2219, Genoa, Italy.

PARTEE B. H. (2010). Privative adjectives : subsective plus coercion. *BÄUERLE, R. et ZIM-MERMANN, TE, éditeurs : Presuppositions and Discourse : Essays Offered to Hans Kamp*, p. 273–285.

SOCHER R., HUVAL B., MANNING C. D. & NG A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 1201–1211, Jeju Island, Korea : Association for Computational Linguistics.

THATER S., DINU G. & PINKAL M. (2009). Ranking paraphrases in context. In *Proceedings of the 2009 Workshop on Applied Textual Inference*, p. 44–47, Suntec, Singapore.

THATER S., FÜRSTENAU H. & PINKAL M. (2010). Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 948–957, Uppsala, Sweden.

TURNEY P. & PANTEL P. (2010). From frequency to meaning : Vector space models of semantics. *Journal of artificial intelligence research*, **37**(1), 141–188.

VAN DE CRUYS T. (2010). A non-negative tensor factorization model for selectional preference induction. *Natural Language Engineering*, **16**(4), 417–437.

ZANZOTTO F. M., KORKONTZELOS I., FALLUCCHI F. & MANANDHAR S. (2010). Estimating linear models for compositional distributional semantics. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, p. 1263–1271, Beijing, China : Coling 2010 Organizing Committee.

Génération de textes : G-TAG revisité avec les Grammaires Catégorielles Abstraites *

Laurence Danlos[1,2,3] Aleksandre Maskharashvili[4, 5, 6] Sylvain Pogodalla[4, 5, 6]
(1) Université Paris Diderot (Paris 7), Paris, F-75013, France
(2) ALPAGE, INRIA Paris–Rocquencourt, Paris, F-75013, France
(3) Institut Universitaire de France, Paris, F-75005, France
(4) INRIA, Villers-lès-Nancy, F-54600, France
(5) Université de Lorraine, LORIA, UMR 7503, Vandœuvre-lès-Nancy, F-54500, France
(6) CNRS, LORIA, UMR 7503, Vandœuvre-lès-Nancy, F-54500, France
{laurence.danlos}{aleksandre.maskharashvili}{sylvain.pogodalla}@inria.fr

Résumé. G-TAG est un formalisme dédié à la génération de textes. Il s'appuie sur les Grammaires d'Arbres Adjoints (TAG) qu'il étend avec des notions propres permettant de construire une forme de surface à partir d'une représentation conceptuelle. Cette représentation conceptuelle est indépendante de la langue, et le formalisme G-TAG a été conçu pour la mise en œuvre de la synthèse dans une langue cible à partir de cette représentation. L'objectif de cet article est d'étudier G-TAG et les notions propres que ce formalisme introduit par le biais des Grammaires Catégorielles Abstraites (ACG) en exploitant leurs propriétés de réversibilité intrinsèque et leur propriété d'encodage des TAG. Nous montrons que les notions clefs d'arbre de g-dérivation et de lexicalisation en G-TAG s'expriment naturellement en ACG. La construction des formes de surface peut alors utiliser les algorithmes généraux associés aux ACG et certaines constructions absentes de G-TAG peuvent être prises en compte sans modification supplémentaire.

Abstract. G-TAG is a formalism dedicated to text generation. It relies on the Tree Adjoining Grammar (TAG) formalism and extends it with several specific notions allowing for the construction of a surface form from a conceptual representation. This conceptual representation is independent from the target language. The goal of this paper is to study G-TAG and its specific notions from the perspective given by Abstract Categorial Grammars (ACG). We use the reversibility property of ACG and the encoding of TAG they offer. We show that the key G-TAG notions of g-derivation tree and lexicalization are naturally expressed in ACG. The construction of surface forms can then rely on the general ACG algorithms and some operations that G-TAG is lacking can be freely accounted for.

Mots-clés : TAG, G-TAG, génération, réalisation syntaxique, grammaires catégorielles abstraites.

Keywords: TAG, G-TAG, generation, syntactic realization, abstract categorial grammars.

1 Introduction

G-TAG (Danlos, 1998; Meunier, 1997; Danlos, 2000) est un formalisme dédié à la génération de textes. Il s'appuie sur les Grammaires d'Arbres Adjoints (TAG) (Joshi *et al.*, 1975; Joshi & Schabes, 1997) qu'il étend avec des notions propres, notamment les arbres de g-dérivation qui permettent de construire une forme de surface (arbre dérivé, ou chaîne de caractères) à partir d'une représentation conceptuelle. Cette représentation conceptuelle est indépendante de la langue, et le formalisme G-TAG a été conçu pour la mise en œuvre de la synthèse dans une langue cible de cette représentation. Ce formalisme a été implanté une première fois en ADA (Meunier, 1997) et plus récemment en .NET, et, dans cette dernière forme, utilisé et commercialisé auprès de Kantar Media, filiale de TNS-Sofres (Meunier *et al.*, 2011; Danlos *et al.*, 2011). Dans cette application, le but est d'accompagner les tableaux retraçant l'évolution des investissements publicitaires des clients par un texte de commentaires synthétisé automatiquement.

Une des motivations pour la définition du formalisme G-TAG était l'observation des différences entre les arbres de dérivation en TAG et les arbres de dépendances sémantiques (Schabes & Shieber, 1994). En analyse, cette observation a

∗. Ce travail a bénéficié d'une aide de l'Agence Nationale de la Recherche portant la référence (ANR-12-CORD-0004).

conduit à des propositions de modification de la notion de structure de dérivation (Kallmeyer, 2002; Joshi *et al.*, 2003; Rambow *et al.*, 2001; Chen-Main & Joshi, 2012). D'autres propositions ont néanmoins montré qu'il était possible de relier aux représentations sémantiques les arbres de dérivation de TAG, sans qu'il ne soit nécessaire de modifier ceux-ci. Soit en utilisant l'unification (Kallmeyer & Romero, 2004, 2007), soit en utilisant une représentation fonctionnelle des TAG (Pogodalla, 2004a, 2007, 2009) dans les Grammaires Catégorielles Abstraites (ACG) (de Groote, 2001). Ces dernières approches présentent l'intérêt d'être exprimées dans un cadre *intrinsèquement réversible*. C'est-à-dire que les grammaires utilisées et la nature des algorithmes mis en œuvre sont les mêmes, que l'on considère le problème de *l'analyse* pour passer de la forme de surface à une représentation sémantique, ou que l'on considère le problème de la *synthèse* pour passer de la représentation sémantique à la forme de surface.

L'objectif de cet article est donc d'étudier G-TAG, les notions propres que ce formalisme introduit, ainsi que ses limitations, par le biais des ACG en exploitant les propriétés de réversibilité de ces dernières et leur propriété d'encodage des TAG. Les ACG permettant d'exprimer également d'autres formalismes de la famille des TAG comme les *multiple component TAG* (MCTAG), certaines limitations de G-TAG peuvent être levées en utilisant ces formalismes sans qu'aucune autre adaptation ne soit nécessaire. Nous montrons que les notions clefs d'arbre de g-dérivation et de lexicalisation s'expriment naturellement en ACG. La construction des formes de surface peut alors utiliser les algorithmes généraux associés aux ACG, permettant d'analyser aussi bien des grammaires non contextuelles que des grammaires légèrement contextuelles (TAG, mais aussi les systèmes de réécriture linéaires non contextuels, LCFRS) (de Groote & Pogodalla, 2004) de manière optimisée (Kanazawa, 2007), et qui sont les mêmes pour l'analyse et la génération. Réciproquement, les principes de conception de G-TAG permettent d'éclairer les éléments opérationnels, notamment liés à la préférence de certaines réalisations, nécessaires à prendre en compte dans les algorithmes des ACG.

2 Génération de textes et G-TAG

L'architecture habituellement considérée dans les processus de génération de textes (Reiter & Dale, 1997) comporte trois sous-processus en cascade, chacun de ces sous-processus étant chargé de la réalisation de différentes tâches. À savoir : la planification du document (ou macro-planification : détermination du contenu, structuration du document), la micro-planification (agrégation, lexicalisation, génération des expressions référentielles), et la réalisation de surface (réalisation linguistique). La première tâche correspond à définir *Que dire ?* tandis que les deux autres réfèrent à *Comment le dire*. G-TAG est dédié à cette seconde tâche. L'entrée du processus de génération associé aux G-TAG, le *Que dire ?* est défini dans une représentation conceptuelle que décrit la section 2.1. Par la suite, le processus comporte trois étapes essentielles : la construction de l'arbre de g-dérivation, la construction de l'arbre g-dérivé, et les traitements ultérieurs pour la finalisation de la forme de surface.

2.1 Représentation conceptuelle

Le langage de représentation conceptuelle utilisé dans G-TAG est essentiellement un langage logique. Il est habituellement présenté sous une forme réifiée de premier ordre, que ce soit dans la logique typée du premier ordre *Login* (Aït-Kaci & Nasr, 1986) qui permet entre autre un contrôle de la bonne formation de la forme conceptuelle et d'abstraire l'ordre et le nombre des arguments par la présence des labels (nom des attributs), ou que ce soit sous forme de structures de représentation du discours segmenté (SDRS) (Danlos *et al.*, 2001).

Dans le présent article, nous adoptons la logique d'ordre supérieur (à la Montague) comme langage de représentation conceptuelle. Cela permet notamment d'éviter les problèmes de quantification implicite sur les labels des objets réifiés et leur traitement compositionnel. Ce faisant, nous considérons :

- que l'ordre et le rôle des arguments des prédicats est pris en compte dans le lien entre arbre de dérivation (ou plutôt le terme qui le représente) et sa représentation conceptuelle (ou sa réalisation sémantique) ;
- que les informations morpho-syntaxiques, habituelles en TAG sous forme de trait, sont pris en compte dans les arbres de dérivations, mais pas dans la représentation conceptuelle ;
- que la génération des expressions référentielles (pronoms, articles définis) est hors du champ de notre proposition. C'est pour l'instant une propriété de l'implantation G-TAG, mais pas du formalisme, qui s'appuie sur la réification. Nous souhaitons pouvoir envisager différentes théories pour la génération de ces expressions.

L'entrée correspondant à (2), qui pourrait engendrer les deux phrases de (3) par exemple, sera donc (1).

$$\text{RWDING}(\text{HUMAN}(\textit{Marie}), \text{HUMAN}(\textit{Jean})) \tag{1}$$

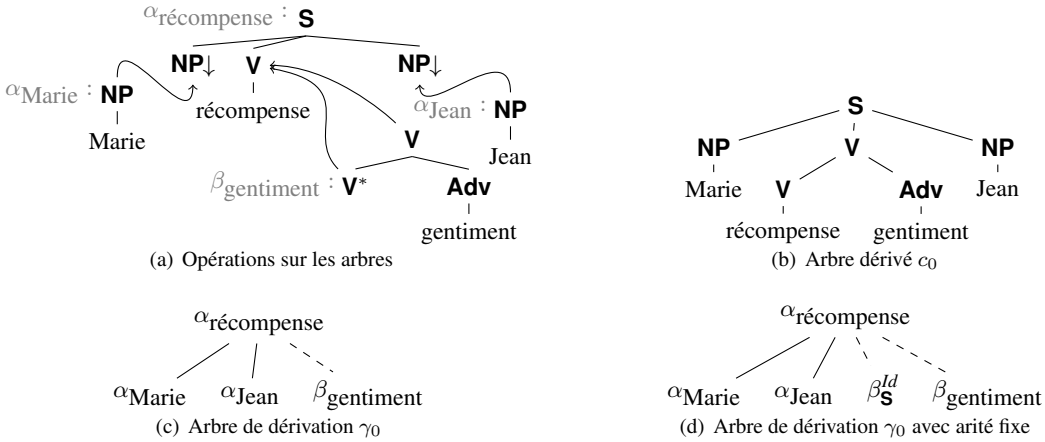
$$
\begin{aligned}
E_{12} &:= \text{RWDING}[\text{RWDER} \mapsto H_2, \text{RWDEE} \mapsto H_1] \\
H_1 &:= \text{HUMAN}[\text{NAME} \mapsto \textit{Jean}, \text{SEX} \mapsto \text{MASC}] \\
H_2 &:= \text{HUMAN}[\text{NAME} \mapsto \textit{Marie}, \text{SEX} \mapsto \text{FEM}]
\end{aligned} \tag{2}
$$

(3) a. Marie a récompensé Jean
b. Jean a été récompensé par Marie

2.2 Arbres de dérivation (TAG)

Les TAG sont des grammaires d'arbres qui mettent en œuvre deux opérations : la substitution et l'adjonction. La substitution permet d'étendre un arbre en remplaçant une feuille étiquetée d'un non terminal par un arbre dont la racine est étiquetée par ce même non terminal. L'adjonction permet *d'insérer* un arbre, appelé *arbre auxiliaire*, possédant un nœud racine et une feuille (nœud pied) étiquetés par un même non terminal. Cet arbre auxiliaire peut être inséré dans un autre arbre à un nœud de même étiquette que la racine de l'arbre auxiliaire. La figure 1(a) montre ces opérations : substitution des nœuds **NP** de l'arbre élémentaire de *récompense*, et adjonction à son nœud **V** de l'arbre auxiliaire de l'adverbe *gentiment*.

Le résultat obtenu est décrit à la figure 1(b), tandis que la figure 1(c) montre la *structure de dérivation*, ou *arbre de dérivation*, qui décrit les opérations effectuées sur les différents arbres :
— les constantes notées $\alpha_{\text{entrée lex}}$ sont les arbres initiaux associés à l'entrée lexicale *entrée lex* ;
— les constantes notées $\beta_{\text{entrée lex}}$ sont les arbres auxiliaires associés à l'entrée lexicale *entrée lex* ;
— un arc plein *parent-enfant* indique que l'arbre enfant a été substitué à une des feuilles de l'arbre parent ;
— un arc parent-enfant en pointillé indique que l'arbre enfant a été adjoint à l'arbre parent en l'un de ses nœuds.



(a) Opérations sur les arbres
(b) Arbre dérivé c_0
(c) Arbre de dérivation γ_0
(d) Arbre de dérivation γ_0 avec arité fixe

FIGURE 1: Analyse TAG de *Marie récompense gentiment Jean*

Remarque. *Dans la définition standard des arbres de dérivation (Vijay-Shanker, 1987), il n'y a pas d'adjonction multiple sur un même nœud, ceci ne changeant pas le pouvoir génératif de la grammaire. Cela signifie notamment qu'un arbre élémentaire peut faire l'objet d'un nombre maximum défini d'adjonctions et de substitutions. Autrement dit, comme étiquette d'un arbre de dérivation (par exemple $\alpha_{récompense}$ dans l'arbre de dérivation de la figure 1(c)), il a une arité* maximum. *Si l'on admet la présence d'éléments unités pour l'opération d'adjonction β_X^{Id} pour une adjonction à un nœud X, on peut même définir très précisément l'arité de ce symbole. L'arbre de la figure 1(c) devrait alors être représenté comme en 1(d).*

Plutôt qu'utiliser les adresses de Gorn pour repérer les nœuds où s'effectuent une opération, on peut fixer un ordre (arbitraire) des arguments et le faire correspondre systématiquement à l'ordre des arguments aussi bien dans la représentation conceptuelle que dans les arbres dérivés.

L'utilisation de la notion étendue de dérivation (Schabes & Shieber, 1994) est également possible dans ce cadre. Elle revient à dédoubler dans les arbres élémentaires chaque nœud pouvant recevoir une adjonction : en un nœud pour les arbres

auxiliaires modifieurs ; et en un autre nœud pour les arbres auxiliaires prédicatifs (selon la terminologie de (Schabes & Shieber, 1994)).

Compte tenu des remarques précédentes, nous insistons sur la convention que nous utiliserons : la notation $\alpha_{\text{entrée lex}}$ est utilisée aussi bien pour représenter l'arbre initial associé à l'entrée lexicale (ou ancré par) *entrée lex* que comme symbole d'arité fixe utilisé pour définir les termes que sont les arbres de dérivation. Si nécessaire, nous indiquerons l'arité du symbole en exposant : $\alpha^4_{\text{récompense}}$. Il en va de même pour la notation $\beta_{\text{entrée lex}}$ à propos des arbres auxiliaires. Alors l'arbre de dérivation de la figure 1(d) s'écrit comme le terme $\alpha^4_{\text{récompense}}(\alpha^0_{\text{Marie}}, \alpha^0_{\text{Jean}}, \beta^{Id}_{\mathbf{S}}, \beta^0_{\text{gentiment}})$ [1].

On constate alors que les arbres de dérivation, comme c'est explicité dans (Schabes & Shieber, 1994), sont des termes *clos*, c'est-à-dire dans lesquels n'apparaissent aucune variable. Or, le processus de génération de (Meunier, 1997) s'appuie sur la correspondance entre le concept à réaliser et les arbres exprimant ce concept pour synthétiser les textes. L'approche adoptée opère de haut en bas (*top-down*) : la relation conceptuelle la plus haute sélectionne un arbre dont les fils dans l'arbre de dérivation seront eux-mêmes générés récursivement par les concepts fils de la relation initiale. Le processus consiste donc à calculer un arbre de dérivation en commençant par la racine. Pour manipuler cet objet en cours de construction, une notion d'arbre de dérivation non complètement instancié est utile. L'arbre de g-dérivation est utilisé à cette fin.

2.3 Arbres de g-dérivation (G-TAG)

Pour exprimer cette notion d'arbre en cours de construction, (Meunier, 1997, Chap. 3, p71) définit les arbres de g-dérivation comme des arbres dont les nœuds sont soit des constantes qui sont des noms d'arbres élémentaires, soit des variables. Deux types de variables sont originellement considérés. D'une part celles qui correspondent au nom des attributs des concepts, utilisés pour associer le rôle sémantique d'un argument et sa position dans l'arbre de dérivation. Et d'autre part celles qui correspondent à l'étiquette qui est valeur de cet attribut (la variable de réification), cette étiquette pouvant être considérée comme à mi-chemin entre la variable de départ et l'arbre de g-dérivation qui la remplacera. Cela correspond à l'évolution de l'arbre de g-dérivation décrite par la figure 2.

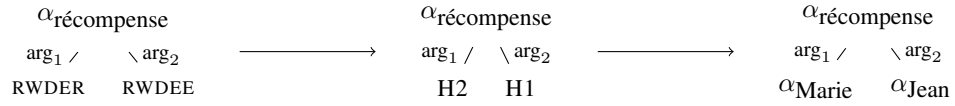
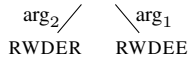


FIGURE 2: Évolution de l'arbre de g-dérivation avec variables vers un arbre instancié

Plus précisément, (Meunier, 1997) définit les constantes apparaissant dans les arbres de g-dérivation comme des noms *d'entrées lexicales* accompagnées de traits, appelés T-traits, plutôt que comme des noms d'arbres élémentaires. Cette distinction, importante pour la mise en œuvre et l'efficacité de G-TAG, permet par exemple de constituer une seule entrée lexicale qui ne distingue les arbres de *récompenser* à la forme active et à la forme passive que par le T-trait [T-trait = ∓passif], réduisant ainsi le nombre de symboles à considérer sans rien changer pour les arguments. En effet, la correspondance entre l'argument d'un concept et l'argument d'une entrée qui le lexicalise est spécifiée par les étiquettes des arcs des arguments qui sont des rôles thématiques (arg_i) qui restent invariants quelle que soit la forme (par ex. active ou passive) de l'entrée lexicale. De plus, l'utilisation de rôles thématiques permet que l'argument conceptuel RWDEE corresponde à l'argument arg_2 des lexicalisations avec le verbe *récompenser*, mais à un argument arg_1 pour une lexicalisation avec *recevoir une récompense* : $\alpha_{\text{recevoir-récompense}}$. Dans l'approche que nous présentons section 3.2,



c'est la manière dont on interprète l'argument de l'arbre de dérivation soit syntaxiquement, soit sémantiquement, qui crée le lien entre le rôle thématique et le rôle conceptuel.

Les T-traits permettent donc de contrôler les constructions syntaxiques possibles. Par exemple, le T-trait [T-trait = +inf] sur l'arbre de g-dérivation associé à la conjonction *pour (que)* spécifie que son deuxième argument doit être une infinitive, ce qui n'est possible que si le sujet vide de la subordonnée infinitive est interprété comme coréférent au sujet de la principale (*Jean passe l'aspirateur pour être récompensé par Marie*). Nous verrons à la section 4 comment modéliser en ACG cette contrainte sur la coréférence des sujets.

1. Pour simplifier les expressions, on fait ici l'hypothèse qu'il n'y a pas d'adjonction possible sur les nœuds **V** et **Adv** de l'adverbe. Sinon, il faudrait écrire le terme $\alpha^4_{\text{récompense}}(\alpha^0_{\text{Marie}}, \alpha^0_{\text{Jean}}, \beta^{Id}_{\mathbf{S}}, \beta^2_{\text{gentiment}}\ \beta^{Id}_{\mathbf{V}}\ \beta^{Id}_{\mathbf{Adv}})$ et l'arbre de dérivation correspondant.

Les termes ou arbres de g-dérivation correspondent donc à des *termes* ou *arbres avec variables*. Pour reprendre les notations habituelles (Comon *et al.*, 2007), si on appelle $\mathcal{F}$ l'ensemble des symboles avec arité, $\mathcal{F} = \bigcup_l \beta_l \cup \bigcup_l \alpha_l$, l'ensemble des arbres de g-dérivations est un sous-ensemble [2] de l'ensemble $T(\mathcal{F}, \mathcal{X})$ des termes construits sur $\mathcal{F}$ et $\mathcal{X}$ un ensemble de variables. Les arbres de dérivations sont les termes clos de cet ensemble. Cette représentation sous-forme d'arbres est isomorphe à celle que nous donnons à l'aide des ACG. Dans cette dernière représentation, les différentes contraintes sont exprimées à l'aides de types.

2.4 Autres notions de G-TAG

Base lexicale La base lexicale associée à un concept est essentiellement un ensemble d'arbres de g-dérivation capables d'exprimer ce concept, notamment parce que les fils de ces arbres expriment les rôles sémantiques associés aux arguments du concept. Cette base lexicale a un rôle opérationnel en permettant de réduire et de diriger la recherche des éléments autorisant la synthèse. Nous la mentionnons ici car elle joue un rôle analogue à la règle Scan dans les analyseurs à chartes. Nous ne donnons pas d'équivalent à cette notion de G-TAG dans la mesure où c'est l'implantation de l'algorithme d'inversion des lexiques dans les ACG, le même pour les tâches d'analyse et de génération, qui met éventuellement en œuvre cette notion si nécessaire [3]. Mais ce n'est pas décrit dans le formalisme lui-même.

Choix lexical Ce choix définit *la meilleure réalisation* (Meunier, 1997, Sect. 3.1.2, p70) d'un concept parmi les éléments de sa base lexicale. Un certain nombre de critères sémantiques, syntaxiques, mais aussi d'interaction entre les choix lexicaux, sont pris en compte à l'aide de différents tests. Quoique ce choix soit très important en pratique, nous considérons ici que toutes les solutions admissibles d'un point de vue syntaxique, c'est-à-dire que la grammaire TAG admettrait, sont possibles. Ce choix ne relève pas des propriétés combinatoires que décrivent la grammaire, et un traitement similaire à celui qui permet la désambiguïsation en syntaxe, avec d'autres critères bien sûr, est envisageable et pourrait bénéficier de toutes les avancées de ce domaine. Il ouvre cependant la question de l'intégration de ces critères aux algorithmes des ACGs, notamment si l'on souhaite qu'ils prennent en compte des informations linguistiques plutôt que statistiques.

Phrase et Discours Un point fort de G-TAG est d'autoriser la génération non de phrases isolées, mais de textes. La grammaire TAG associée contient donc des arbres élémentaires avec des signes de ponctuation, notamment le point. Ces arbres sont associés à des représentations conceptuelles exprimant la relation sémantique discursive entre deux phrases. Ainsi, les trois textes de (4) expriment la même représentation conceptuelle : SUCC(EAT(HUMAN JEAN), LEAVE(HUMAN MARIE)). Un certain nombre de contraintes sont exprimées par des traits, par exemple pour signifier que dans les constructions mentionnées, seul *ensuite* permet de coordonner deux phrases distinctes. Nous ne les utilisons pas ici pour simplifier les formules. Mais ils ne posent aucun problème théorique.

(4) a. Jean mange. Ensuite, Marie part.
b. Jean mange avant que Marie ne parte.
c. Jean mange avant le départ de Marie.

3 Génération et analyse dans les ACG

3.1 Généralités sur les ACG

Définitions et compositions Une ACG définit deux langages qu'elle met en relation : un *langage abstrait*, qui peut être vu comme un ensemble abstrait de structures grammaticales, et un *langage objet*, représentant les formes réalisées des structures abstraites. Ici, le langage abstrait correspond à la structure grammaticale que l'on veut manipuler, c'est-à-dire l'arbre de dérivation. Il sera mis en relation à l'aide d'un premier *lexique* avec le langage objet des arbres dérivés (ou des chaînes de caractère), et à l'aide d'un deuxième lexique avec le langage objet des représentations conceptuelles.

Definition 1 (Signature d'ordre supérieur)**.** *Une* signature d'ordre supérieur *est un triplet* $\Sigma = \langle A, C, \tau \rangle$ *où :*
— A *est un ensemble de types atomiques ;*
— C *est un ensemble fini de constantes ;*

2. Certains termes n'étant pas possibles du fait des contraintes exprimées soit par les représentations conceptuelles, soit par les arbres dérivés.
3. Voir par exemple (Kanazawa, 2007) et son extension à une stratégie d'analyse particulière (Kanazawa, 2008).

— $\tau : C \to \mathcal{T}(A)$ *assigne à chaque constante de* C *un type de* $\mathcal{T}(A)$ *où* $\mathcal{T}(A) ::= A | \mathcal{T}(A) \multimap \mathcal{T}(A)$ [4].
On appelle $\Lambda(\Sigma)$ *l'ensemble des* λ*-termes que l'on peut construire avec la signature* Σ *et* $t : \alpha$ *signifie que le terme* t *a le type* α.

Ainsi, pour obtenir les arbres de la figure 1, nous pouvons définir une première signature :

$$\Sigma_{\text{dérivations}} = \begin{cases} A_{\Sigma_{\text{dérivations}}} = \{\mathbf{NP}, \mathbf{V}, \mathbf{S}, \mathbf{Adv}, \mathbf{V_A}, \mathbf{S_A}, \ldots\} \\ C_{\Sigma_{\text{dérivations}}} = \{C_{\textit{récompense}}, C_{\textit{Marie}}, C_{\textit{gentiment}}, C_{\textit{Jean}}, \beta^{Id}_{\mathbf{S}}, \beta^{Id}_{\mathbf{V}} \ldots\} \\ C_{\textit{Marie}} : \mathbf{NP} \qquad C_{\textit{récompense}} : \mathbf{S_A} \to \mathbf{V_A} \to \mathbf{NP} \to \mathbf{NP} \to \mathbf{S} \\ C_{\textit{Jean}} : \mathbf{NP} \qquad C_{\textit{gentiment}} : \mathbf{V_A} \\ \beta^{Id}_{\mathbf{V}} : \mathbf{V_A} \qquad \beta^{Id}_{\mathbf{S}} : \mathbf{S_A} \end{cases}$$

Cette signature permet de construire les structures de dérivation que sont les arbres de dérivation. Les types $X_{\mathbf{A}}$ correspondent aux types des arbres prêts à être adjoints. Pour le détail de l'encodage systématique d'une TAG dans une ACG, nous renvoyons le lecteur à (de Groote, 2002; Pogodalla, 2004a, 2007, 2009).

Une autre signature nous permet de construire les arbres dérivés. Cette signature ne comporte qu'un seul type, le type τ des arbres.

$$\Sigma_{\text{dérivés}} = \begin{cases} A_{\Sigma_{\text{dérivés}}} = \{\tau\} \\ C_{\Sigma_{\text{dérivés}}} = \{\mathbf{S}_2, \mathbf{S}_3, \mathbf{V}_1, \mathbf{V}_2, \mathbf{Adv}_1, \mathbf{NP}_1, \mathbf{NP}_2, \textit{Marie}, \textit{Jean}, \textit{gentiment}, \textit{récompense}, \ldots\} \\ \textit{Marie}, \textit{Jean}, \textit{récompense}, \textit{gentiment} : \tau \\ \mathbf{NP}_1, \mathbf{V}_1, \mathbf{Adv}_1 : \tau \to \tau \\ \mathbf{NP}_2, \mathbf{S}_2, \mathbf{V}_2 : \tau \to \tau \to \tau \\ \mathbf{S}_3 : \tau \to \tau \to \tau \to \tau \end{cases}$$

Chaque constante permet de construire un arbre, de type τ. Elles sont distinguées par leur arité ($\mathbf{NP}_1$, $\mathbf{NP}_2$, etc.). Cela correspond aux différentes arités effectivement exprimées dans les arbres élémentaires de la grammaire TAG mais qui sont généralement laissées implicites.

Definition 2 (Lexique)**.** *Étant données une signature d'ordre supérieur* $\Sigma_1 = \langle A_1, C_1, \tau_1 \rangle$ *et une signature d'ordre supérieur* $\Sigma_2 = \langle A_2, C_2, \tau_2 \rangle$*, un* lexique := de Σ_1 vers Σ_2 *est défini par la donnée de* $:\overset{\tau}{=}$ *et* $:\overset{c}{=}$ *tels que :*

— $:\overset{\tau}{=} \; : A_1 \to \mathcal{T}(A_2)$ *est une fonction d'interprétation des types atomiques de* Σ_1 *comme des types implicatifs construits à partir de* A_2*. On appellera* $:\overset{\tau}{=}$ *également son extension homomorphique à tous les types de* $\mathcal{T}(A_1)$ *;*
— $:\overset{c}{=} \; : C_1 \to \Lambda(\Sigma_2)$ *est une fonction d'interprétation des constantes de* Σ_1 *comme des* λ*-termes construits à partir de* Σ_2*. On appellera* $:\overset{c}{=}$ *également son extension homomorphique à tous les termes de* $\Lambda(\Sigma_1)$ *;*
— *les fonctions d'interprétation sont compatibles avec la relation de typage, c'est-à-dire que pour tout* $c \in C_1$ *et* $t : \alpha \in \Lambda(\Sigma_2)$ *tels que* $c :\overset{c}{=} t$*, alors* $\tau_1(c) :\overset{\tau}{=} \alpha$ *(le type de l'image de* c *est l'image du type de* c*).*

Dans la suite, on utilisera sans ambiguïté := *pour* $:\overset{\tau}{=}$ *ou* $:\overset{c}{=}$*. On définit également la fonction d'interprétation* $[\![\cdot]\!]$ *telle que pour tout* $t \in \Lambda(\Sigma_1)$ *et* $t := u$*,* $[\![t]\!] = u$.

Nous pouvons maintenant définir un lexique qui relie les termes abstraits de $\Lambda(\Sigma_{\text{dérivations}})$ aux termes objets de $\Lambda(\Sigma_{\text{dérivés}})$. Les tables 1 et 2 définissent bien un lexique. Le premier donne l'interprétation des types atomiques. On remarquera avec l'interprétation des types $\mathbf{V_A}$ et $\mathbf{S_A}$ que l'interprétation d'un type atomique peut être un type non atomique. Cela explicite le fait que l'adjonction est une fonction qui prend un (sous-)arbre et retourne un arbre qui sera à son tour inséré. La deuxième table donne l'interprétation des constantes. Pour l'interprétation de $C_{\textit{récompense}}$ (et des verbes en général), S correspond à l'arbre auxiliaire qui peut éventuellement s'adjoindre au nœud $\mathbf{S}$ et a à l'arbre auxiliaire qui peut éventuellement s'adjoindre au nœud $\mathbf{V}$, tandis que s et o correspondent aux arbres $\mathbf{NP}$ qui seront respectivement sujet et objet.

$\mathbf{NP}$	$\mathbf{S}$	$\mathbf{V}$	$\mathbf{V_A}$	$\mathbf{S_A}$
$:=_{\text{syntaxe}} \tau$	$:=_{\text{syntaxe}} \tau$	$:=_{\text{syntaxe}} \tau$	$:=_{\text{syntaxe}} \tau \to \tau$	$:=_{\text{syntaxe}} \tau \to \tau$

TABLE 1: Interprétation des types de $\Sigma_{\text{dérivations}}$ vers $\Sigma_{\text{dérivés}}$

Definition 3 (Grammaire catégorielle abstraite)**.** *Une* grammaire catégorielle abstraite *est un quadruplet* $\mathcal{G} = \langle \Sigma_1, \Sigma_2, :=, s \rangle$ *où :*

4. $\multimap$ est l'implication linéaire. Les variables abstraites par les λ ne peuvent être utilisées qu'une et une seule fois.

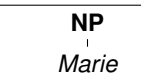
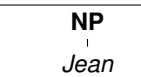
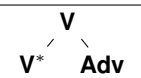
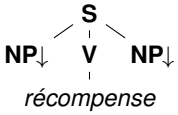
Interprétation des constantes		Arbre dérivé TAG correspondant
$C_{\textit{Marie}}$	$:=_{\text{syntaxe}} \mathbf{NP}_1\ \textit{Marie}$	
$C_{\textit{Jean}}$	$:=_{\text{syntaxe}} \mathbf{NP}_1\ \textit{Jean}$	
β^{Id}_X	$:= \lambda x.x \quad (X \in \{\mathbf{V_A}, \mathbf{S_A}, \ldots\})$	
$C_{\textit{gentiment}}$	$:=_{\text{syntaxe}} \lambda x.\mathbf{V}_2\ x\ (\mathbf{Adv}_1\ \textit{gentiment})$	
$C_{\textit{récompense}}$	$:=_{\text{syntaxe}} \lambda S\ a\ s\ o.S\ (\mathbf{S}_3\ s\ (a\ (\mathbf{V}_1\ \textit{récompense}))\ o)$	

TABLE 2: Lexique reliant les arbres de dérivation aux arbres dérivés

— Σ_1 *est une signature d'ordre supérieur, et* Σ_2 *une signature d'ordre supérieur. Elles sont appelés* vocabulaire abstrait *et* vocabulaire objet *;*
— $:= \ : \Sigma_1 \to \Sigma_2$ *est un lexique ;*
— s *est un type atomique du vocabulaire abstrait, appelé le* type distingué *de la grammaire.*

Definition 4 (Langages abstrait et objet)**.** *Soit* $\mathcal{G} = \langle \Sigma_1, \Sigma_2, :=, s \rangle$ *une grammaire catégorielle abstraite.*

1. *Le* langage abstrait $\mathcal{A}(\mathcal{G})$ *engendré par* $\mathcal{G}$ *est défini par* $\mathcal{A}(\mathcal{G}) = \{t \in \Lambda(\Sigma_1) | t : s\}$
2. *Le* langage objet $\mathcal{O}(\mathcal{G})$ *engendré par* $\mathcal{G}$ *est défini par* $\mathcal{O}(\mathcal{G}) = \{t \in \Lambda(\Sigma_2) | \exists u \in \mathcal{A}(\mathcal{G})\ \textit{avec}\ u := t\}$

Le terme correspondant à l'arbre de dérivation de la figure 1(c) est le terme : $t_0 = C_{\textit{récompense}}\ \beta^{Id}_{\mathbf{S}}\ C_{\textit{gentiment}}\ C_{\textit{Marie}} C_{\textit{Jean}}$. Il est bien typé, de type $\mathbf{S}$. Il appartient donc bien au langage abstrait de l'ACG $\mathcal{G}_{\text{syntaxe}}$ dont le lexique est donné dans les tables 1 et 2. On peut donc calculer son image par le lexique. Par définition, elle appartiendra au langage objet :

$$t_0 :=_{\text{syntaxe}} \mathbf{S}_3\ (\mathbf{NP}_1\ \textit{Marie})\ (\mathbf{V}_2\ (\mathbf{V}_1 \textit{récompense})\ (\mathbf{Adv}_1\ \textit{gentiment}))\ (\mathbf{NP}_1\ \textit{Jean})$$

La définition des ACG permet de considérer différents types d'architecture. On peut par exemple composer deux ACG de sorte que le vocabulaire objet de l'une soit également le vocabulaire abstrait de l'autre. C'est le cas si l'on veut considérer le lien entre les arbres dérivés, cette fois vus comme un langage abstrait, et leur production (*yield*) comme langage de chaînes. On utilisera une nouvelle ACG $\mathcal{G}_{\text{surface}}$ qui aura comme vocabulaire abstrait $\Sigma_{\text{dérivés}}$ le vocabulaire objet de $\mathcal{G}_{\text{syntaxe}}$ et comme vocabulaire objet la signature Σ_{string} dont le seul type est le type des chaînes de caractère σ, qui possède comme constantes l'opération de concaténation, l'élément vide, et les chaînes *Marie*, *Jean*, *récompense*, ... En utilisant le lexique défini dans la table 3 on obtient la chaîne de caractères associée à t_0 :

$$\begin{aligned} t_0 &:=_{\text{syntaxe}} \mathbf{S}_3\ (\mathbf{NP}_1\ \textit{Marie})\ (\mathbf{V}_2\ (\mathbf{V}_1 \textit{récompense})\ (\mathbf{Adv}_1\ \textit{gentiment}))\ (\mathbf{NP}_1\ \textit{Jean}) \\ &:=_{\text{surface} \circ \text{syntaxe}} \textit{Marie} + \textit{récompense} + \textit{gentiment} + \textit{Jean} \end{aligned}$$

Cette composition d'ACG est illustrée dans la partie gauche de la figure 3.

Analyse dans les ACG On peut maintenant préciser ce que l'on entend par problème d'analyse dans les ACG. Soit une ACG $\mathcal{G} = \langle \Sigma_1, \Sigma_2, :=, s \rangle$. Analyser un terme $u \in \Lambda(\Sigma_2)$, c'est-à-dire un terme construit sur le vocabulaire objet, revient à *trouver un terme* t *tel que* $t \in \mathcal{A}(\mathcal{G})$ *et* $t := u$. Il s'agit donc *d'inverser le lexique*.

Les propriétés des ACG dites d'ordre 2 [5] ont été particulièrement étudiées. Elles permettent d'encoder les formalismes faiblement contextuelles comme les TAG, les systèmes de réécriture linéaires non contextuels, les grammaires non contextuelles multiples (de Groote & Pogodalla, 2004; Salvati, 2006; Kanazawa, 2009) et la complexité de l'analyse est polynomiale (Kanazawa, 2008).

Or ces résultats ne dépendent que de l'ordre de la signature abstraite, mais pas du vocabulaire objet. L'inversion du morphisme reste possible dans les mêmes conditions y compris si le vocabulaire objet permet de construire des formules logiques pour la représentation conceptuelle. C'est ce qui permet de qualifier les ACG d'intrinsèquement réversibles.

5. Ce sont les ACG dont les types des constantes abstraites sont au plus d'ordre 2, avec $\text{ord}(a) = 1$ si a est un type atomique et $\text{ord}(\alpha \to \beta) = \max(\text{ord}(\beta), \text{ord}(\alpha) + 1)$. Autrement dit, ce sont les ACG dont les structures de dérivations, les termes abstraits, sont des arbres.

τ	$:=_{\text{surface}}\ \sigma$	*Marie*	$:=_{\text{surface}}\ \textit{Marie}$
Jean	$:=_{\text{surface}}\ \textit{Jean}$	*récompense*	$:=_{\text{surface}}\ \textit{récompense}$
gentiment	$:=_{\text{surface}}\ \textit{gentiment}$	$\mathbf{NP}_1, \mathbf{V}_1, \mathbf{Adv}_1$	$:=_{\text{surface}}\ \lambda x.x$
$\mathbf{NP}_2, \mathbf{S}_2, \mathbf{V}_2$	$:=_{\text{surface}}\ \lambda x\ y.x + y$	$\mathbf{S}_3$	$:=_{\text{surface}}\ \lambda x\ y\ z.x + y + z$

TABLE 3: Lexique pour les formes de surfaces à partir des formes dérivées

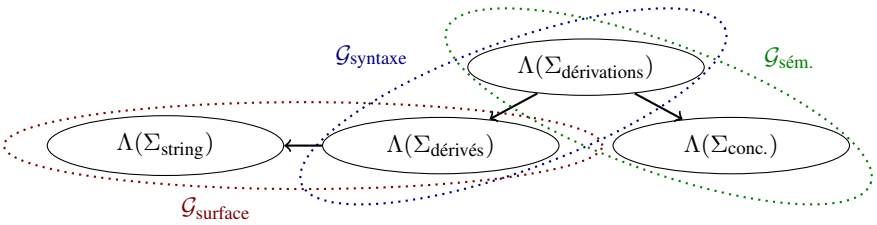


FIGURE 3: Architecture ACG utilisée pour les TAG et leur interface syntaxe-sémantique

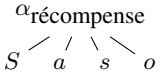
Interface syntaxe-sémantique La réalisation de l'interface syntaxe-sémantique avec les ACG s'obtient par un mode de composition des ACG différent du précédent. Il s'agit de partager la structure de dérivation, autrement dit de considérer deux ACG qui partagent un même vocabulaire abstrait. La partie droite de la figure 3 illustre cette composition. Le vocabulaire abstrait $\Sigma_{\text{dérivations}}$ étant déjà défini, il nous suffit de définir le vocabulaire objet $\Sigma_{\text{conc.}}$ et le lexique de $\mathcal{G}_{\text{sém.}}$. $\Sigma_{\text{conc.}}$ comprend les deux types atomiques THING (entités) et PROP (propositions), qui correspondent respectivement aux types e et t chez Montague, ainsi que les constantes typées de la table 4. Le lecteur pourra vérifier que :

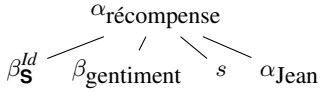
$$t_0 :=_{\text{sém.}} \text{KINDLY}(\text{RWDING}(\text{HUMAN MARIE}, \text{HUMAN JEAN}))$$

$\mathbf{S}$	$:=_{\text{sém.}}\ \text{PROP}$	$C_{\textit{Marie}}$	$:=_{\text{sém.}}\ \lambda P.P\ (\text{HUMAN MARIE})$
$\mathbf{N}$	$:=_{\text{sém.}}\ (\text{THING} \to \text{PROP}) \to \text{PROP}$	$C_{\textit{Jean}}$	$:=_{\text{sém.}}\ \lambda P.P\ (\text{HUMAN JEAN})$
$\mathbf{V_A}$	$:=_{\text{sém.}}\ \text{PROP} \to \text{PROP}$	$\beta^{Id}_{\mathbf{V}}$	$:=_{\text{sém.}}\ \lambda x.x$
$\mathbf{S_A}$	$:=_{\text{sém.}}\ \text{PROP} \to \text{PROP}$	$\beta^{Id}_{\mathbf{S}}$	$:=_{\text{sém.}}\ \lambda x.x$
$C_{\textit{gentiment}}$	$:=_{\text{sém.}}\ \lambda v\ s.\text{KINDLY}(v\ s)$	$C_{\textit{récompense}}$	$:=_{\text{sém.}}\ \lambda S\ a\ s\ o.S(s(\lambda x.o(\lambda y.a\ (\text{RWDING}(x, y)))))$

TABLE 4: Lexique de $\mathcal{G}_{\text{sém.}}$ pour la représentation conceptuelle

3.2 TAG, G-TAG, et ACG

Cette architecture permet donc d'utiliser un vocabulaire abstrait partagé entre deux ACG pour réaliser l'interface entre syntaxe et sémantique. Or, dans le cas de l'encodage des TAG avec les ACG, le langage abstrait des structures de dérivation, c'est-à-dire les termes de type $\mathbf{S}$, correspond aux arbres de dérivation des TAG. Par ailleurs, on a vu dans la section 2.3 que les arbres de g-derivation de G-TAG sont des arbres de dérivation TAG non complètement instanciés. Cela correspond en fait aux termes de $\Lambda(\Sigma_{\text{dérivations}})$ d'ordre 2. Ainsi, le terme $C_{\textit{récompense}} : \mathbf{S_A} \to \mathbf{V_A} \to \mathbf{NP} \to \mathbf{NP} \to \mathbf{S}$ peut être écrit sous-forme η-longue $\lambda S\ a\ s\ o.C_{\textit{récompense}}\ S\ a\ s\ o$. Il correspond à l'abre de g-dérivation (à l'ordre des arguments près : la modélisation TAG en ACG place habituellement les adjonctions en début, alors que les TAG les placent à la fin) :  . La même correspondance peut s'observer sur des termes et des arbres partiellement instanciés.

Par exemple entre le terme $\lambda s.C_{\textit{récompense}}\ \beta^{Id}_{\mathbf{S}}\ C_{\textit{gentiment}}\ s\ C_{\textit{Jean}}$ et l'arbre  où l'argument

correspondant au sujet n'est pas encore instancié.

$\mathcal{G}_{\text{sém.}}$ permet également d'exprimer la notion de base lexicale d'un concept C. L'ensemble des arbres de g-dérivation capables d'exprimer ce concept peut être défini comme l'ensemble des constantes du vocabulaire abstrait dont l'image par $:=_{\text{sém.}}$ admet le concept C comme sous-terme : $BL(\text{C}) = \{c \in C_{\Sigma_{\text{dérivations}}} \mid c{:=_{\text{sém.}}}\ u$ et C est un sous-terme de $u\}$.

Montrer l'équivalence formelle entre les arbres de g-dérivation et les termes de $\Lambda(\Sigma_{\text{dérivations}})$ d'ordre 2 irait au-delà de cet article. Mais l'approche que nous proposons consiste donc à *utiliser l'architecture des ACG pour construire les arbres de dérivations (ou arbres de g-dérivation complètement instanciés) à partir des termes représentant les formes conceptuelles*. Cela nous permet d'utiliser les propriétés de réversibilité des ACG de second ordre pour construire ces arbres de dérivations, ainsi que les techniques d'optimisation qui y sont liées.

Cette approche nous permet également de dépasser la limitation de G-TAG concernant les verbes ponts (verbes à complétive permettant l'extraction hors de celle-ci), généralement représentés par des arbres auxiliaires en TAG. Dans les dérivations, ils apparaissent donc comme dépendants de la tête de la complétive, alors que conceptuellement leur prédicat a portée sur le verbe de la complétive. En G-TAG, il a été choisi de ne pas modéliser ces verbes qui n'apparaissent pas dans les textes techniques (Danlos, 1998, 2000). On voit bien la difficulté technique liée à l'inversion de l'ordre des arguments entre l'arbre de g-dérivation et la représentation conceptuelle. Les études formelles sur les ACG ont montré que ce problème apparent n'en était pas vraiment un à l'ordre 2 [6] tant que les lexiques sont presque linéaires (*almost linear*) [7]. Comme il a été montré par ailleurs que l'arbre de dérivation permet d'exprimer de manière adéquate ces inversions de portée (Pogodalla, 2004b,a), il n'est plus nécessaire de faire des hypothèses sur la grammaire TAG utilisée.

4 Exemples

Les exemples de cette section exprimés au second ordre utilisent une ACG et l'algorithme d'analyse général du toolkit ACG [8]. Le premier exemple a pour objectif d'illustrer le lien qui doit être fait entre les arguments du terme abstrait et ses réalisations syntaxique et sémantique. Nous étendons donc $\mathcal{G}_{\text{syntaxe}}$ et $\mathcal{G}_{\text{sém.}}$ avec les interprétations des tables 5 et 6. Nous laissons le lecteur vérifier que :

$$C_{\textit{récompense}}\ \beta^{Id}_{\mathbf{S}}\ \beta^{Id}_{\mathbf{V}}\ C_{\textit{Marie}}\ C_{\textit{Jean}} :=_{\text{sém.}} \textsc{rwding}(\textsc{human Marie}, \textsc{human Jean})$$
$$C_{\textit{être récompensé}}\ \beta^{Id}_{\mathbf{S}}\ \beta^{Id}_{\mathbf{V}}\ C_{\textit{Jean}}\ C_{\textit{Marie}} :=_{\text{sém.}} \textsc{rwding}(\textsc{human Marie}, \textsc{human Jean})$$
$$C_{\textit{donne récompense}}\ \beta^{Id}_{\mathbf{S}}\ \beta^{Id}_{\mathbf{V}}\ C_{\textit{Marie}}\ C_{\textit{Jean}} :=_{\text{sém.}} \textsc{rwding}(\textsc{human Marie}, \textsc{human Jean})$$
$$C_{\textit{reçoit récompense}}\ \beta^{Id}_{\mathbf{S}}\ \beta^{Id}_{\mathbf{V}}\ C_{\textit{Jean}}\ C_{\textit{Marie}} :=_{\text{sém.}} \textsc{rwding}(\textsc{human Marie}, \textsc{human Jean})$$

Les constantes abstraites $C_{\textit{pour que}}$ et $C_{\textit{pour}}$ de la table 5 permettent d'engendrer les textes de (5) [9] à partir de la représentation conceptuelle : $\textsc{goal}(\textsc{vacc}(\textsc{human jean}), \textsc{rwding}(\textsc{human Marie}, \textsc{human Jean}))$. Il est à noter que $C_{\textit{pour}}$, à strictement parler, ne correspond pas à un arbre TAG mais permet la combinaison de plusieurs d'entre eux pour former l'expression p_1 *pour* p_2 où p_1 est le résultat de la substitution de **NP**↓ dans s_1 par le **NP** donné en paramètre, tandis que p_2 est le résultat de la substitution du **NP**↓ dans s_2 par l'arbre **NP** / **Pro** [10] (syntaxiquement la chaîne vide alors que sémantiquement le sujet est le même que celui donné en argument à s_1 pour former p_1). Cela permet notamment d'assurer que le sujet est bien partagé sémantiquement. Cela requiert simplement un nouveau type **Sws** (et des règles lexicales) dont les interprétations sont telles que **Sws**$:=_{\text{syntaxe}} \tau \to \tau$ et **Sws**$:=_{\text{sém.}} [\![\mathbf{NP}]\!] \to \textsc{Prop}$ avec **NP**$:=_{\text{sém.}} [\![\mathbf{NP}]\!] = (\textsc{Thing} \to \textsc{Prop}) \to \textsc{Prop}$. (5d) n'est pas engendré car il n'y a pas de constante abstraite pour le passif de *donner une récompense* qui n'est pas passivable en français. Or, comme l'interprétation sémantique de $C_{\textit{pour}}$ requiert que les deux propositions aient même sujet syntaxique, pour que le sujet de la deuxième soit le patient du prédicat RWDING, il faut un passif.

(5) a. Jean passe l'aspirateur pour que Marie récompense Jean
b. Jean passe l'aspirateur pour être récompensé par Marie
c. Jean passe l'aspirateur pour que Marie donne une récompense à Jean
d. *Jean passe l'aspirateur pour être donné une récompense par Marie

6. En fait, on peut lier cela à une complexité qui reste polynomiale mais dont le coefficient du polynôme croît avec la complexité du lexique exprimée par l'ordre maximum des termes réalisant les constantes abstraites. C'est ce qui explique la différence de complexité de l'analyse pour les grammaires non contextuelles et les grammaires TAG. Si des bornes maximales sont connues dans le cas des langages de chaînes et d'arbres, ce n'est pas le cas pour des langages objets en général.
7. C'est-à-dire qu'aucune variable n'est effacée (λ-abstraction vide) et que seules les variables de type atomique peuvent apparaître plusieurs fois. Dans le cas général, cela reste décidable mais extrêmement complexe (Salvati, 2010) et lève la restriction de monotonie sémantique (Shieber, 1988).
8. http://www.loria.fr/equipes/calligramme/acg/#Software
9. Nous donnons les exemples sans pronom puisque nous ne les traitons pas pour l'instant, mais ils seraient nécessaires.
10. Pour des raisons de simplicité, nous omettons les traits morpho-syntaxiques tels que subjonctif ou infinitif.

Le dernier exemple que nous souhaitons évoquer concerne *ensuite*. La constante abstraite que nous avons définie permet d'attacher l'adverbe à la proposition. (Danlos, 2000) indique que l'obtention de l'attachement au **V** passe par un post-traitement, ou alors requiert un formalisme plus expressif que TAG. Nous pouvons exprimer ceci grâce à l'ordre supérieur avec une constante $C'_{\textit{ensuite}} : \mathbf{S} \rightarrow (\mathbf{V_A} \rightarrow \mathbf{S}) \rightarrow \mathbf{S}$ dont les interprétations sont : $C'_{\textit{ensuite}} :=_{\text{sém.}} \lambda s_1\ s_2.\textsc{succ}(s_1, s_2\ (\lambda x.x))$ et $C'_{\textit{ensuite}} :=_{\text{syntaxe}} \lambda s_1\ s_2.\mathbf{S}_2\ s_1\ (s_2(\lambda x.\mathbf{V}_2\ x\ \textit{ensuite}))$. Nous laissons le lecteur vérifier que les termes t_1 et t_2 donnés en (6) dérivent bien la même représentation sémantique $\textsc{succ}(\textsc{vacc}(\textsc{human Jean}), \textsc{rwding}(\textsc{human Marie}, \textsc{human Jean}))$ et engendrent respectivement les deux phrases de (7).

$$
\begin{aligned}
t_1 &= C_{\textit{ensuite}}(C_{\textit{passe l'aspirateur}}\ \beta^{Id}_{\mathbf{S}}\ \beta^{Id}_{\mathbf{V}} C_{\textit{Jean}})\ (C_{\textit{donne récompense}}\ \beta^{Id}_{\mathbf{S}}\ \beta^{Id}_{\mathbf{V}}\ C_{\textit{Marie}}\ C_{\textit{Jean}}) \\
t_2 &= C'_{\textit{ensuite}}(C_{\textit{passe l'aspirateur}}\ \beta^{Id}_{\mathbf{S}}\ \beta^{Id}_{\mathbf{V}} C_{\textit{Jean}})\ (\lambda v.C_{\textit{donne récompense}}\ \beta^{Id}_{\mathbf{S}}\ v\ C_{\textit{Marie}}\ C_{\textit{Jean}})
\end{aligned}
\tag{6}
$$

(7) a. Jean passe l'aspirateur. Ensuite Marie donne une récompense à Jean.
b. Jean passe l'aspirateur. Marie donne ensuite une récompense à Jean.

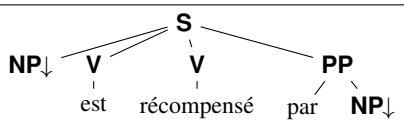
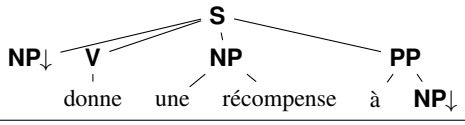
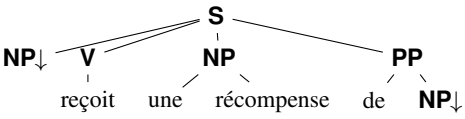
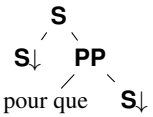
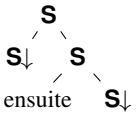
$C_{\textit{être récompensé}}$	$: \mathbf{S_A} \rightarrow \mathbf{V_A} \rightarrow \mathbf{NP} \rightarrow \mathbf{NP} \rightarrow \mathbf{S}$	
$C_{\textit{donne récompense}}$	$: \mathbf{S_A} \rightarrow \mathbf{V_A} \rightarrow \mathbf{NP} \rightarrow \mathbf{NP} \rightarrow \mathbf{S}$	
$C_{\textit{reçoit récompense}}$	$: \mathbf{S_A} \rightarrow \mathbf{V_A} \rightarrow \mathbf{NP} \rightarrow \mathbf{NP} \rightarrow \mathbf{S}$	
$C_{\textit{pour que}}$	$: \mathbf{S} \rightarrow \mathbf{S} \rightarrow \mathbf{S}$	
$C_{\textit{pour}}$	$: \mathbf{Sws} \rightarrow \mathbf{Sws} \rightarrow \mathbf{NP} \rightarrow \mathbf{S}$	
$C_{\textit{ensuite}}$	$: \mathbf{S} \rightarrow \mathbf{S} \rightarrow \mathbf{S}$	

TABLE 5: Arbres élémentaires et constantes abstraites pour les différentes expressions du concept RWDING

$C_{\textit{être récompensé}}$	$:=_{\text{syntaxe}} \lambda S\ a\ s\ o.S\ (\mathbf{S}_4\ (\mathbf{NP}_1\ s)\ (a\ (\mathbf{V}_2\ \textit{est}))\ (\mathbf{V}_1\ \textit{récompensé})\ (\mathbf{PP}_2\ \textit{par}\ o))$ $:=_{\text{sém.}} \lambda S\ a\ s\ o.S(a(\textsc{rwding}(o, s)))$
$C_{\textit{donne récompense}}$	$:=_{\text{syntaxe}} \lambda S\ a\ s\ o.S\ (\mathbf{S}_4\ (\mathbf{NP}_1\ s)\ (a\ (\mathbf{V}_1\ \textit{donne}))\ (\mathbf{NP}_2\ \textit{une récompense})\ (\mathbf{PP}_2\ \textit{à}\ o))$ $:=_{\text{sém.}} \lambda S\ a\ s\ o.S(a(\textsc{rwding}(s, o)))$
$C_{\textit{reçoit récompense}}$	$:=_{\text{syntaxe}} \lambda S\ a\ s\ o.S\ (\mathbf{S}_4\ (\mathbf{NP}_1\ s)\ (a\ (\mathbf{V}_1\ \textit{reçoit}))\ (\mathbf{NP}_2\ \textit{une récompense})\ (\mathbf{PP}_2\ \textit{de}\ o))$ $:=_{\text{sém.}} \lambda S\ a\ s\ o.S(a(\textsc{rwding}(o, s)))$
$C_{\textit{pour que}}$	$:=_{\text{syntaxe}} \lambda s_1\ s_2.\mathbf{S}_2\ s_1\ (\mathbf{PP}_2\ (\textit{pour que})\ s_2)$ $:=_{\text{sém.}} \lambda s_1\ s_2.\textsc{goal}(s_1, s_2)$
$C_{\textit{pour}}$	$:=_{\text{syntaxe}} \lambda s_1\ s_2\ s.\mathbf{S}_2\ (s_1\ s)\ (\mathbf{PP}_2\ \textit{pour}\ (s_2\ (\mathbf{NP}_1\ \mathbf{Pro})))$ $:=_{\text{sém.}} \lambda s_1\ s_2\ n.\textsc{goal}(s_1\ n, s_2\ n)$
$C_{\textit{ensuite}}$	$:=_{\text{syntaxe}} \lambda s_1\ s_2.\mathbf{S}_2\ s_1\ (\mathbf{PP}_2\ (\textit{ensuite})\ s_2)$ $:=_{\text{sém.}} \lambda s_1\ s_2.\textsc{succ}(s_1, s_2)$

TABLE 6: Différentes expressions du concept RWDING

5 Conclusion et perspectives

Nous avons étudié G-TAG et ses notions propres, notamment l'arbre de g-dérivation, et montré que ce dernier correspondait aux termes abstraits d'une ACG de second ordre. Cela nous permet : d'une part d'utiliser les propriétés de réversibilité intrinsèque de ce formalisme ; d'autre part de généraliser l'approche à des phénomènes non traités (verbes ponts) tout en restant dans les TAG ; enfin de créer facilement des liens vers des formalismes plus expressifs (cf l'attachement de *ensuite* au nœud **V**). Cependant, un certain nombre de traitements propres aux implantations G-TAG ne sont pas encore traduits ici. La génération des expressions référentielles en est un. Une autre limitation concerne la parallélisation, ou la factorisation de certains évènements dans certaines constructions. En effet, la représentation conceptuelle que nous avons adoptée associe la représentation SUCC(GOAL(VACC(HUMAN JEAN), RWDING(HUMAN MARIE, HUMAN JEAN), NAP(HUMAN JEAN)) au texte *Jean a passé l'aspirateur pour être récompensé par Marie. Ensuite il a fait la sieste*. Or le prédicat GOAL ne devrait pas être dans la portée de SUCC. Ceci est traité à l'aide d'opérations particulières en G-TAG (Meunier, 1997, Section 6.1.2). Nous souhaitons éviter les opérations spécifiques et pour cela utiliser dans des travaux ultérieurs la représentation sémantique d'ordre supérieur décrite par (Danlos, 2009). Par ailleurs, le problème de guider le choix des réalisations lexicales, notamment par des règles et des connaissances linguistiques comme en G-TAG, et d'indiquer des préférences aux algorithmes utilisés dans les ACG est ouvert. Enfin, la clarification du statut de l'arbre de g-dérivation du point de vue des propriétés formelles des langages engendrés nous permet d'envisager de comparer plus précisément cette approche à celles qui considèrent les arbres de dérivation comme les arbres engendrés par une grammaire régulière d'arbres (Schmitz & Le Roux, 2008) dont les dérivations sont utilisées comme pivot pour la génération (Gardent & Perez-Beltrachini, 2010).

Références

AÏT-KACI H. & NASR R. (1986). LOGIN : A logic programming language with built-in inheritance. *The Journal of logic programming*, **3**(3), 185–215. doi :10.1016/0743-1066(86)90013-0.

CHEN-MAIN J. & JOSHI A. K. (2012). A dependency perspective on the adequacy of tree local multi-component tree adjoining grammar. *Journal of Logic and Computation*. doi :10.1093/logcom/exs012.

COMON H., DAUCHET M., GILLERON R., LÖDING C., JACQUEMARD F., LUGIEZ D., TISON S. & TOMMASI M. (2007). Tree Automata Techniques and Applications. Available on : http://www.grappa.univ-lille3.fr/tata. Release October, 12th 2007.

DANLOS L. (1998). G-TAG : Un formalisme lexicalisé pour la génération de textes inspiré de TAG. *Traitement Automatique des Langues*, **39**(2). http://hal.inria.fr/inria-00098489.

DANLOS L. (2000). G-TAG : A lexicalized formalism for text generation inspired by Tree Adjoining Grammar. In A. ABEILLÉ & O. RAMBOW, Eds., *Tree Adjoining Grammars : Formalisms, Linguistic Analysis, and Processing*, volume 107 of *CSLI Lecture Notes*, p. 343–370 : CSLI Publications.

DANLOS L. (2009). D-STAG : un formalisme d'analyse automatique de discours basé sur les TAG synchrones. *T.A.L.*, **50**(1), 111–143. http://hal.inria.fr/inria-00524743/en/.

DANLOS L., GAIFFE B. & ROUSSARIE L. (2001). Document sructuring à la SDRT. In H. HORACEK, N. NICOLOV & L. WANNER, Eds., *Proceedings of the ACL 2001 Eighth European Workshop on Natural Language Generation (EWNLG)*. http://aclweb.org/anthology/W/W01/W01-0803.pdf.

DANLOS L., MEUNIER F. & COMBET V. (2011). EasyText : an operational NLG system. In *ENLG 2011, 13th European Workshop on Natural Language Generation*. http://hal.inria.fr/inria-00614760/en/.

DE GROOTE P. (2001). Towards Abstract Categorial Grammars. In *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter, Proceedings of the Conference*, p. 148–155. http://aclweb.org/anthology/P/P01/P01-1033.pdf.

DE GROOTE P. (2002). Tree-Adjoining Grammars as Abstract Categorial Grammars. In *TAG+6, Proceedings of the sixth International Workshop on Tree Adjoining Grammars and Related Frameworks*, p. 145–150 : Università di Venezia. http://www.loria.fr/equipes/calligramme/acg/publications/2002-tag+6.pdf.

DE GROOTE P. & POGODALLA S. (2004). On the expressive power of Abstract Categorial Grammars : Representing context-free formalisms. *Journal of Logic, Language and Information*, **13**(4), 421–438. doi :10.1007/s10849-004-2114-x.

GARDENT C. & PEREZ-BELTRACHINI L. (2010). RTG based surface realisation for TAG. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, p. 367–375, Beijing, China : Coling 2010 Organizing Committee. http://www.aclweb.org/anthology/C10-1042.

JOSHI A. K., KALLMEYER L. & ROMERO M. (2003). Flexible Composition in LTAG : Quantifier Scope and Inverse Linking. In H. BUNT, I. VAN DER SLUIS & R. MORANTE, Eds., *Proceedings of IWCS-5*.

JOSHI A. K., LEVY L. S. & TAKAHASHI M. (1975). Tree Adjunct Grammars. *Journal of Computer and System Sciences*, **10**(1), 136–163. doi :10.1016/S0022-0000(75)80019-5.

JOSHI A. K. & SCHABES Y. (1997). Tree-adjoining grammars. In G. ROZENBERG & A. SALOMAA, Eds., *Handbook of formal languages*, volume 3, chapter 2. Springer.

KALLMEYER L. (2002). Using an Enriched TAG Derivation Structure as Basis for Semantics. In *Proceedings of TAG+6*.

KALLMEYER L. & ROMERO M. (2004). LTAG Semantics with Semantic Unification. In *Seventh International Workshop on Tree Adjoining Grammar and Related Formalisms - TAG+7*, p. 155–162. http://www.sfs.uni-tuebingen.de/~lk/papers/kallmrom-tag+7.pdf.

KALLMEYER L. & ROMERO M. (2007). Scope and Situation Binding for LTAG. *Research on Language and Computation*, **6**(1), 3–52. doi :10.1007/s11168-008-9046-6.

KANAZAWA M. (2007). Parsing and Generation as Datalog Queries. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*, p. 176–183, Prague, Czech Republic : Association for Computational Linguistics. http://www.aclweb.org/anthology/P/P07/P07-1023.

KANAZAWA M. (2008). A prefix-correct Earley recognizer for multiple context-free grammars. In *Proceedings of the Ninth International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+9)*, p. 49–56, Tuebingen, Germany.

KANAZAWA M. (2009). Second-Order Abstract Categorial Grammars as Hyperedge Replacement Grammars. *Journal of Logic, Language, and Information*, **19**(2), 137–161. doi :10.1007/s10849-009-9109-6.

MEUNIER F. (1997). *Implantation du formalisme de génération G-TAG*. PhD thesis, Université Paris 7 — Denis Diderot.

MEUNIER F., DANLOS L. & COMBET V. (2011). EasyText : un système opérationnel de génération de textes. In *Actes de la 18e conférence sur le Traitement Automatique des Langues Naturelles*, Montpellier, France. http://hal.inria.fr/inria-00607708.

POGODALLA S. (2004a). Computing Semantic Representation : Towards ACG Abstract Terms as Derivation Trees. In *Seventh International Workshop on Tree Adjoining Grammar and Related Formalisms - TAG+7*, p. 64–71, Vancouver, BC, Canada. http://hal.inria.fr/inria-00107768.

POGODALLA S. (2004b). Vers un statut de l'arbre de dérivation : exemples de construction de représentations sémantiques pour les Grammaires d'Arbres Adjoints. In *Traitement Automatique des Langues Naturelles - TALN'04*, p. 10 p, Fès, Maroc : none. http://hal.inria.fr/inria-00107767.

POGODALLA S. (2007). Ambiguïté de portée et approche fonctionnelle des TAG. In *Traitement Automatique des Langues Naturelles - TALN 2007*, p. 10 p., Toulouse, France. http://hal.inria.fr/inria-00141913.

POGODALLA S. (2009). Advances in Abstract Categorial Grammars : Language Theory and Linguistic Modeling. ESSLLI 2009 Lecture Notes, Part II. http://hal.inria.fr/hal-00749297.

RAMBOW O., VIJAY-SHANKER K. & WEIR D. (2001). D-Substitution Grammars. *Computational Linguistics*, **27**(1), 87–121. http://aclweb.org/anthology/J/J01/J01-1004.pdf.

REITER E. & DALE R. (1997). Building applied natural language generation systems. *Natural Language Engineering*, **3**(1), 57–87. doi :10.1017/S1351324997001502.

SALVATI S. (2006). Encoding second order string ACG with Deterministic Tree Walking Transducers. In SHULY WINTNER, Ed., *Proceedings of The 11th conference on Formal Grammar FG 2006*, FG Online Proceedings, p. 143–156, Malaga Espagne : CSLI Publications. http://cslipublications.stanford.edu/FG/2006/salvati.pdf.

SALVATI S. (2010). On the membership problem for non-linear Abstract Categorial Grammars. *Journal of Logic, Language and Information*, **19**(2), 163–183. doi :10.1007/s10849-009-9110-0.

SCHABES Y. & SHIEBER S. M. (1994). An alternative conception of tree-adjoining derivation. *Computational Linguistics*, **20**(1), 91–124. http://aclweb.org/anthology/J/J94/J94-1004.pdf.

SCHMITZ S. & LE ROUX J. (2008). Feature Unification in TAG Derivation Trees. In C. GARDENT & A. SARKAR, Eds., *Proceedings of the 9th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+'08)*, p. 141–148, Tübingen, Germany. http://arxiv.org/abs/0804.4584.

SHIEBER S. M. (1988). A Uniform Architecture for Parsing and Generation. In *Proceedings of the 12th International Conference on Computational Linguistics*, volume 2, p. 614–619, Budapest. http://dash.harvard.edu/bitstream/handle/1/2265286/Shieber_UniformArchitecture.pdf.

VIJAY-SHANKER K. (1987). *A Study of Tree Adjoining Grammars*. PhD thesis, University of Pennsylvania.

Apprentissage partiellement supervisé d'un étiqueteur morpho-syntaxique par transfert cross-lingue

Guillaume Wisniewski[1, 2] Nicolas Pécheux[1, 2] Elena Knyazeva[1,2] Alexandre Allauzen[1,2]
François Yvon[2]
(1) Université Paris Sud, 91 403 Orsay CEDEX
(2) LIMSI-CNRS, 91 403 Orsay CEDEX
{nom.prénom}@limsi.fr

Résumé. Les méthodes de transfert cross-lingue permettent partiellement de pallier l'absence de corpus annotés, en particulier dans le cas de langues peu dotées en ressources linguistiques. Le transfert d'étiquettes morpho-syntaxiques depuis une langue riche en ressources, complété et corrigé par un dictionnaire associant à chaque mot un ensemble d'étiquettes autorisées, ne fournit cependant qu'une information de supervision incomplète. Dans ce travail, nous reformulons ce problème dans le cadre de l'*apprentissage ambigu* et proposons une nouvelle méthode pour apprendre un analyseur de manière faiblement supervisée à partir d'un modèle à base d'historique. L'évaluation de cette approche montre une amélioration sensible des performances par rapport aux méthodes de l'état de l'art pour trois langues sur quatre considérées, avec des gains jusqu'à 3,9% absolus ou 35,8% relatifs.

Abstract. When Part-of-Speech annotated data is scarce, e.g. for under resourced languages, one can turn to cross-lingual transfer and crawled dictionaries to collect partially supervised data. We cast this problem in the framework of *ambiguous learning* and show how to learn an accurate history-based model. This method is evaluated on four languages and yields improvements over state-of-the-art for three of them, with gains up to 3.9% absolute or 35.8% relative.

Mots-clés : apprentissage partiellement supervisé, analyse morpho-syntaxique, transfert cross-lingue.

Keywords: Weakly Supervised Learning, Part-of-Speech Tagging, Cross-Lingual Transfer.

1 Introduction

Les catégories morpho-syntaxiques, qui regroupent les mots partageant un même comportement syntaxique et/ou morphologique, constituent une source d'information pertinente pour de nombreuses tâches de traitement automatique des langues (TAL). Elles sont par exemple aujourd'hui presque systématiquement calculées en prétraitement pour des tâches d'extraction d'information, pour la reconnaissance d'entités nommées ou encore en traduction automatique, sans parler de leur utilisation en analyse syntaxique. Étant donné leur importance, de nombreux travaux se sont attachés à prédire automatiquement ces étiquettes en utilisant une grande variété de méthodes d'apprentissage supervisé. Ces méthodes atteignent aujourd'hui un niveau de performances proche de celui d'un annotateur humain, du moins lorsqu'elles sont entrainées sur des corpus annotés suffisamment grands dans le domaine d'intérêt (Manning, 2011).

L'annotation manuelle d'un corpus reste cependant un processus complexe, fastidieux et onéreux qui nécessite une solide expertise linguistique (Abeillé *et al.*, 2003), même si les outils aujourd'hui disponibles peuvent aider à accélérer très significativement cette démarche (Garrette & Baldridge, 2013). Il n'existe donc actuellement des corpus annotés avec des informations morpho-syntaxiques que pour un nombre de langues et de domaines réduits. Différentes approches ont été proposées dans la littérature pour réduire cet effort d'annotation (voire pour s'en passer complètement) afin de développer des analyseurs morpho-syntaxiques pour des langues et des domaines pour lesquels ces ressources n'existent pas.

Un premier type de solution consiste à estimer des classes de mots automatiquement à partir de corpus non annotés, en

regroupant les unités qui possèdent un même comportement distributionnel ; ces classes doivent ensuite être projetées sur les catégories morpho-syntaxiques traditionnelles pour pouvoir être interprétées. Une grande variété de méthodes ont été proposées dans la littérature pour réaliser cette tâche, depuis (Brown *et al.*, 1992) jusqu'aux travaux plus récents de (Banko & Moore, 2004; Toutanova & Johnson, 2007). Malgré des progrès constants, leurs performances restent en général trop faibles pour permettre leur utilisation dans des applications de TAL (Christodoulopoulos *et al.*, 2010). Cette approche peut être largement améliorée dès lors que l'on dispose d'une poignée de données annotées en plus des données non étiquetées (apprentissage semi-supervisé) : les annotations serviront, par exemple, à initialiser et/ou à désambigüiser les catégories apprises automatiquement.

Il est également possible, pour projeter les mots sur une liste de catégories prédéfinies, d'utiliser des dictionnaires qui contraignent la liste des étiquettes possibles de chaque mot. Ces dictionnaires, qui permettent de réaliser une désambigüisation partielle, s'avèrent très utiles (par exemple dans un cadre de modèle à données latentes), lorsque de grands corpus non annotés sont disponibles pour l'apprentissage (Merialdo, 1994; Banko & Moore, 2004). De tels dictionnaires peuvent aujourd'hui être obtenus automatiquement à relativement bas cout (Li *et al.*, 2012), par exemple à partir des données de projets tels que `Wiktionary` [1].

Le transfert cross-lingue offre une autre manière, complémentaire, de contourner l'absence ou la rareté de données annotées. Le principe du transfert cross-lingues est d'exploiter des corpus de textes parallèles, qui peuvent aujourd'hui être collectés automatiquement en grande quantité (Resnik & Smith, 2003) et d'utiliser ceux-ci pour *transférer* les sorties des outils d'analyse appliqués à une langue *source* riche en données annotées vers une langue *cible* moins bien dotée. Ainsi, en exploitant les alignements automatiques au niveau des mots, il est possible de projeter les étiquettes morpho-syntaxiques des phrases sources vers les phrases cibles (Yarowsky *et al.*, 2001). Dans la lignée de (Das & Petrov, 2011; Li *et al.*, 2012), Täckström *et al.* (2013) a montré qu'il était possible d'apprendre des analyseurs morpho-syntaxiques de bonne qualité de cette manière, si l'information extraite à partir des alignements venait compléter les indications extraites d'un dictionnaire associant à chaque mot un ensemble d'étiquettes morpho-syntaxiques autorisées. Wang & Manning (2014) montrent qu'il est également possible et préférable de transférer les probabilités calculées par les outils d'analyse en langue source plutôt que de projeter uniquement les étiquettes prédites. La méthode de transfert des étiquettes entre langue source et langue cible, ainsi que l'extraction et l'utilisation des dictionnaires, sont détaillées dans la partie 2.

Les deux approches proposées par Täckström *et al.* (2013), permettant d'apprendre à partir de ces deux sources de données (les étiquettes projetées et les dictionnaires), reposent sur des modèles de séquences (HMM et CRF) et sur une généralisation *ad hoc* de leur critère d'apprentissage, afin d'intégrer les différentes sources d'information. Dans ce travail, nous proposons de reformuler le problème du transfert cross-lingue dans le cadre de l'*apprentissage ambigu* (Bordes *et al.*, 2010; Cour *et al.*, 2011) dont l'objectif est d'estimer un classifieur lorsque le système ne peut accéder, lors de la phase d'apprentissage, qu'à un ensemble d'étiquettes possibles dont une seule est juste et non à l'étiquette de référence. À partir des résultats théoriques développés dans (Bordes *et al.*, 2010), nous introduisons une méthode d'apprentissage capable d'apprendre un étiqueteur morpho-syntaxique dans un contexte faiblement supervisé. Ce modèle d'apprentissage est décrit dans la partie 3 et son évaluation sur quatre langues est présentée dans la partie 4.

Le code source et l'ensemble des ressources utilisées dans ce travail sont disponibles à l'url `http://perso.limsi.fr/wisniews/ambiguous`.

2 Création de corpus d'apprentissage par transfert d'étiquettes

L'objectif de ce travail est de développer des étiqueteurs morpho-syntaxiques en s'appuyant sur le transfert d'annotations entre phrases parallèles afin de pouvoir complètement se dispenser, lors de l'apprentissage, de données étiquetées manuellement. Le transfert d'annotations nécessite de définir une correspondance entre étiquettes des langues source et cible, correspondance qui est obtenue dans ce travail en utilisant un ensemble universel d'étiquettes morpho-syntaxiques simples décrit dans la sous-section 2.1. En suivant la méthode proposée par Täckström *et al.* (2013), nous utilisons deux sources complémentaires d'information pour déterminer les étiquettes des différents mots de la langue cible par transfert cross-lingue : un dictionnaire associant à un mot-type donné l'ensemble de ses étiquettes possibles (partie 2.2) et les alignements entre une phrase annotée et sa traduction (partie 2.3). Ces informations sont fusionnées pour étiqueter automatiquement un corpus d'apprentissage (partie 2.4).

1. `http://www.wiktionary.org/`

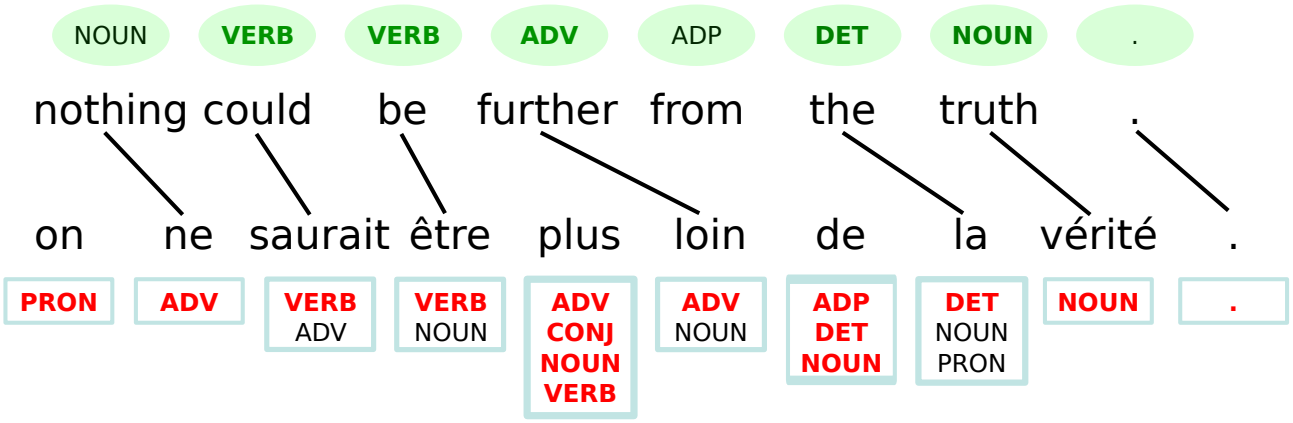


FIGURE 1 – *Exemple de transfert d'étiquettes d'une phrase source (haut) en anglais vers une phrase cible (bas) en français, extrait du corpus d'apprentissage. Pour chaque mot cible, les étiquettes autorisées par les contraintes de type sont représentées dans le cadre bleu. Les étiquettes morpho-syntaxiques de la phrase source sont transférées vers la phrase cible uniquement lorsque celles-ci sont « compatibles » avec les contraintes de type (elles sont dans ce cas représentées en vert). Au final, les étiquettes en rouge constituent l'information (ambigüe) de supervision de la phrase cible.*

2.1 Un ensemble universel d'étiquettes morpho-syntaxique

La possibilité de transférer l'information morpho-syntaxique d'une langue à un autre suppose que cette information puisse être décrite de la même manière dans les deux langues. Même si cette hypothèse forte est hautement controversée (Evans & Levinson, 2009; Broschart, 2009), Petrov *et al.* (2012) définissent 12 étiquettes morpho-syntaxiques à gros grain choisies en raison de leur « universalité » (les catégories identifiées sont relativement stables d'une langue à l'autre) et de leur utilité dans une chaine de traitement de TAL. Ces étiquettes universelles sont les suivantes : NOUN (noms), VERB (verbes), ADJ (adjectifs), ADV (adverbes), PRON (pronoms), DET (déterminants et articles), ADP (prépositions et postpositions), NUM (numéraux), CONJ (conjonctions), PRT (particules), « . » (symboles de ponctuations) et X (pour tout ce qui échappe aux autres catégories, comme par exemple les abréviations ou les mots étrangers). Ces catégories sont uniquement décrites par des exemples et par leur association à des corpus existants et n'ont pas vraiment fait l'objet d'une caractérisation formelle. Par la suite, nous supposerons toujours que toutes les étiquettes morpho-syntaxiques ont été transformées en étiquettes universelles.

2.2 Utilisation de dictionnaires

La première source d'information utilisée pour prédire les informations morpho-syntaxiques est appelée *contrainte de type* et est constituée par un dictionnaire qui associe à chaque mot-type l'ensemble des étiquettes autorisées pour ce mot. La figure 1 donne un exemple d'étiquettes autorisées pour une phrase en français. Les mots « on », « ne », « vérité » et « . » sont entièrement désambigüisés par le dictionnaire. Comme expliqué dans la partie 2.4, ces contraintes permettent de réduire les étiquettes possibles pour chaque mot et de filtrer les annotations transférées en suivant les liens d'alignements.

Plusieurs types de dictionnaires peuvent être utilisés pour déterminer les étiquettes possibles pour un mot. Dans ce travail, nous utilisons des dictionnaires extraits automatiquement de `Wiktionary` en utilisant les méthodes et les heuristiques introduites par Li *et al.* (2012). `Wiktionary` est un dictionnaire de grande envergure, collaboratif et libre et peut être considéré comme une source relativement fiable d'information. Chacune de ses entrées contient les définitions, les étiquettes morpho-syntaxiques et des informations de prononciation, cela pour un grand nombre de mots [2]. Li *et al.* (2012) étudient en détail la couverture et l'exactitude des étiquettes morpho-syntaxiques extraites de `Wiktionary`. Sur les corpus annotés manuellement considérés dans nos expériences, nous observons que ces contraintes de type sont exactes pour plus de 94% des mots-occurrences (voir section 4.1 pour plus de détails). Il est important de noter que les informations de `Wiktionary` sont données en forme libre et que leur extraction et leur conversion vers l'ensemble universel d'étiquettes est une tâche fastidieuse.

2. Par exemple, les dictionnaires français et grec utilisés dans nos expériences contiennent respectivement 1 242 728 et 21 857 entrées. Ces entrées décrivent aussi bien des stemmes que des formes fléchies.

2.3 Transfert cross-lingue d'étiquettes

La deuxième source d'information, appelée *contrainte d'occurrence (token constraint)*, utilise les liens d'alignements [3], lorsqu'ils existent, pour projeter l'étiquette d'un mot-occurrence source sur un mot-occurrence cible. La figure 1 montre un exemple d'alignement entre une phrase source et une phrase cible. L'alignement manquant entre « from » et « de » ne permet pas de transférer l'étiquette ADP. Pour limiter le bruit lié aux erreurs des alignements automatiques, nous calculons au préalable pour chaque mot-type la distribution des étiquettes qui seraient transférées par cette méthode. Pour un mot-occurrence donné, le transfert d'étiquette n'est finalement pris en compte que si cette étiquette transférée est l'une des deux étiquettes les plus fréquentes pour ce mot-type [4]. De plus nous utilisons cette information pour compléter le dictionnaire de types : lorsqu'un mot-type n'est pas dans le dictionnaire extrait de `Wiktionary`, nous utilisons comme contraintes de type ces deux étiquettes les plus fréquentes [5]. Par exemple, sur la figure 1 le verbe conjugué « saurait » n'est pas une entrée de `Wiktionary` et se voit attribuer les deux étiquettes VERB et ADV

Täckström *et al.* (2013) décrivent de manière détaillée l'impact de ces deux types de contraintes et montrent que chacune d'elles apporte des informations complémentaires.

2.4 Prise en compte des deux sources complémentaires d'information

Les deux sources d'information introduites précédemment sont fusionnées en utilisant les règles décrites par l'algorithme 1, qui s'inspire de la méthode de (Täckström *et al.*, 2013). La figure 1 donne un exemple de transfert et de filtrage des étiquettes d'une phrase source vers une phrase cible.

Après transfert et filtrage des étiquettes, les mots cibles sont donc associés à un ensemble d'étiquettes (en rouge, figure 1) et non à une unique étiquette de référence. Un mot-occurrence cible peut cependant être associé à une unique étiquette, dans le cas du transfert d'une étiquette ou dans le cas où la contrainte de type est réduite à une étiquette. La partie 4.1 montre que c'est le cas pour environ 80% des mots-occurrences. Dans la partie suivante, nous expliquons comment il est possible d'entrainer un analyseur morpho-syntaxique n'utilisant que cette *information ambigüe* comme supervision.

Algorithme 1: Règles utilisées pour transférer les étiquettes à partir d'une phrase source.

input : mot w, d dictionnaire décrivant les contraintes de type et un alignement entre les phrases source et cible
output : l'ensemble des étiquettes possibles pour le mot w
$occurrence \leftarrow$ {étiquette du mot avec lequel w est aligné} ; `// ∅ si w n'est pas aligné`
$type \leftarrow d[w]$
if $type \cap occurrence \neq \emptyset$ **then**
  return *occurrence;*
else
  return *type*;
end

3 Modèles de séquences pour l'apprentissage faiblement supervisé

Pour apprendre un modèle de séquences dans un cadre faiblement supervisé, nous utilisons un modèle à base d'historique (Black *et al.*, 1992; Collins, 2003; Tsuruoka *et al.*, 2011) avec une méthode d'apprentissage proche de LaSO (Daumé & Marcu, 2005). Dans les modèles à base d'historique, la prédiction d'une structure complexe (ici la séquence d'étiquettes morpho-syntaxiques) est modélisée sous la forme d'une suite de problèmes de décision, consistant chacun à prédire l'étiquette d'une observation. Chaque décision est prise par un classifieur multi-classe utilisant comme descripteurs des informations extraites de la structure d'entrée, ainsi que les décisions prises antérieurement (c'est-à-dire une sortie partiellement désambigüisée). Ces modèles permettent donc de *réduire* l'apprentissage structuré en un problème d'apprentissage multi-classe.

3. Nous ne considérons que des alignements `1:1` entre mots sources et cibles.
4. Il est aussi possible d'effectuer ce filtrage en seuillant la distribution comme dans (Täckström *et al.*, 2013), mais nous n'avons pas observé de différences entre ces deux heuristiques.
5. Dans les rares cas où un mot-type n'est jamais aligné dans le corpus d'entrainement, nous utilisons le jeu complet d'étiquettes. Dans nos expériences c'est le cas pour moins de 0,2% des mots-occurrences.

Notons $\mathbf{x} = (x_i)_{i=1}^{n}$ la séquence d'observations et $\mathcal{Y}$ l'ensemble des étiquettes possibles (dans notre cas les 12 étiquettes universelles). L'inférence consiste à prédire les étiquettes les unes après les autres en utilisant, ici, un modèle linéaire :

$$y_i^* = \arg\max_{y \in \mathcal{Y}} \langle \mathbf{w} | \phi(\mathbf{x}, i, y, h_i) \rangle \tag{1}$$

où $\langle a|b \rangle$ dénote le produit scalaire de a et b, y_i^* est l'étiquette prédite pour la i-ème observation, $\mathbf{w}$ le vecteur de poids, $h_i = y_1^*, ..., y_{i-1}^*$ l'historique décrivant les décisions passées à l'étape i et ϕ un vecteur de traits représentant de manière jointe l'observation, l'étiquette candidate et l'historique. Ainsi, l'inférence peut être vue comme une recherche gloutonne dans l'espace des $\#\{\mathcal{Y}\}^n$ étiquettes possibles pour la séquence observée. Ce type de modèle, qui sacrifie un optimum global au prix d'une plus grande flexibilité des traits [6], a été utilisé avec succès dans de nombreuses applications de TAL (Kazama & Torisawa, 2007; Ratinov & Roth, 2009; Tsuruoka *et al.*, 2011).

L'apprentissage, comme décrit par l'algorithme 2, consiste à effectuer successivement l'inférence pour chaque séquence d'entrée et à corriger le vecteur de poids chaque fois qu'une décision erronée est prise. De manière cruciale (Wolpert, 1992; Ross & Bagnell, 2010), lors de l'apprentissage, les historiques doivent être constitués des étiquettes prédites par le modèle jusque-là, et non des étiquettes de références comme dans (Daumé & Marcu, 2005), afin de rester en cohérence avec la situation qui sera rencontrée au moment du décodage. Cette particularité est la principale différence avec la méthode originale de (Daumé & Marcu, 2005).

L'utilisation d'un modèle à base d'historique permet d'apprendre facilement un modèle de séquences à partir d'une information ambigüe : l'information de supervision disponible est utilisée pour déterminer si une décision est bonne ou erronée ce qui, comme nous allons le montrer dans les deux paragraphes suivants, permet d'adapter la méthode d'apprentissage à un contexte supervisé ou ambigu.

3.1 Apprentissage (fortement) supervisé

Lorsque l'apprentissage est supervisé, la séquence d'étiquettes correcte est connue. Il est donc possible de savoir, à chaque étape de l'inférence, si l'étiquette prédite est différente de l'étiquette de référence. Dès que c'est le cas, la décision est considérée comme erronée et le vecteur de poids est mis à jour, comme pour un perceptron, de la manière suivante :

$$\mathbf{w} \leftarrow \mathbf{w} + \phi\left(\mathbf{x}, i, \hat{y}_i, h_i\right) - \phi\left(\mathbf{x}, i, y_i^*, h_i\right) \tag{2}$$

où y_i^* et $\hat{y}_i$ sont, respectivement, l'étiquette prédite et l'étiquette de référence. Cette mise à jour correspond à un pas de descente de gradient stochastique et permet de renforcer le score de l'étiquette de référence par rapport à tous les autres.

3.2 Apprentissage ambigu (ou faiblement supervisé)

Lorsque la séquence d'étiquettes de référence n'est pas connue, il est tout de même possible d'apprendre un modèle de séquences si l'on dispose pour chaque observation d'un sous-ensemble d'étiquettes possibles, noté $\hat{\mathcal{Y}}_i$. Dans ce cas, une décision est considérée erronée à partir du moment où l'étiquette prédite n'est pas dans l'ensemble des étiquettes autorisées par l'information de supervision ambigüe. Le vecteur de poids est alors mis à jour comme suit :

$$\mathbf{w} \leftarrow \mathbf{w} + \sum_{\hat{y}_i \in \hat{\mathcal{Y}}_i} \left(\phi\left(\mathbf{x}, i, \hat{y}_i, h\right) - \phi\left(\mathbf{x}, i, y_i^*, h_i\right)\right) \tag{3}$$

Cette mise à jour vise à renforcer toutes les étiquettes de $\hat{\mathcal{Y}}_i$.

Dans le cadre de l'apprentissage ambigu (Bordes *et al.*, 2010; Cour *et al.*, 2011), il est possible de montrer en faisant des hypothèses peu restrictives (qui reviennent à dire, en première approximation, qu'il suffit que l'étiquette correcte soit présente dans l'ensemble des étiquettes possibles et qu'elle n'y soit pas systématiquement associée à une autre étiquette), qu'un classifieur entrainé uniquement à partir d'informations de supervision ambigüe revient à un classifieur appris à partir de l'information de supervision complète [7]. Nous nous limiterons ici à donner l'intuition de ce résultat sur un exemple

6. La complexité de l'apprentissage et de l'inférence ne dépend pas de la taille de l'historique, alors que considérer des modèles comme les CRFs avec des dépendances dont l'ordre est supérieur à deux rend aussi bien la complexité de l'apprentissage que l'inférence prohibitive.

7. Bordes *et al.* (2010) définissent une fonction de perte dite *ambigüe (ambiguous loss)*, qui est optimisée par des mises à jour semblables à celles données par l'équation (3) et montrent que la solution qui permet d'obtenir l'erreur minimale pour cette fonction de perte est également la solution du problème minimisant la perte 0/1 que l'on pourrait évaluer si l'on connaissait l'étiquette de référence.

jouet : considérons un corpus contenant deux phrases, « `la souris` » et « `la féline` » dont les étiquettes ambigües sont, respectivement $[\{\text{DET}\}, \{\text{VERB}, \text{NOUN}\}]$ et $[\{\text{DET}\}, \{\text{NOUN}, \text{ADJ}\}]$ et considérons les deux traits correspondant au mot et au mot précédent ; lors de l'application de la règle de mise à jour donnée par l'équation (3), le trait décrivant le mot précédent (« la » dans les deux cas) associé à l'étiquette NOUN sera « renforcé » deux fois, contre une pour les étiquettes incorrectes (ADJ et VERB). Ce « partage de l'information » par l'intermédiaire des traits permet que l'étiquette NOUN soit correctement prédite lors de l'inférence, même si elle n'a jamais été associée aux deux noms de manière non ambigüe. De manière plus générale, tant que deux étiquettes ne sont pas systématiquement associées dans les ensembles de supervision, la répétition des mises à jour renforcera plus souvent la « bonne » étiquette et, au final, celle-ci finira par avoir le plus grand score.

Algorithme 2: Algorithme d'apprentissage. Dans le cas ambigu, $\hat{\mathcal{Y}}_i$ est l'ensemble des étiquettes autorisées ; dans le cas supervisé, cet ensemble est réduit à l'étiquette de référence. Le nombre T d'itérations effectuées, est un hyperparamètre de l'algorithme.

for $t \in [\![1, T]\!]$ **do**
  Tirer au hasard un exemple $\mathbf{x}, \hat{\mathbf{y}}$;
  $h \leftarrow$ liste vide ; `// Initialise un historique vide`
  for $i \in [\![1, n]\!]$ **do**
    $y_i^* = \arg\max_{y \in \mathcal{Y}} \langle \mathbf{w} | \phi(\mathbf{x}, i, y, h_i) \rangle$;
    if $y_i^* \notin \hat{\mathcal{Y}}_i$ **then**
      $\mathbf{w} \leftarrow$ mise_à_jour$(\mathbf{w}, \mathbf{x}, i, \hat{\mathcal{Y}}_i, y_i^*, h_i)$; `// Suivant les Équations (2) et (3)`
    end
    ajouter(y_i^*, h);
  end
end

4 Expériences

4.1 Corpus

Nous considérons quatre langues [8] pour évaluer notre approche : le grec, le français, l'espagnol et l'allemand. Dans toutes ces expériences, nous partons de l'anglais comme langue source et utilisons comme données parallèles les corpus EUROPARL et NEWSCOMMENTARY [9]. Pour chaque paire de langues considérée, les corpus sont alignés en utilisant la chaine de traitement standard de MOSES (Koehn *et al.*, 2007) avec l'heuristique d'intersection pour fusionner les deux directions d'alignements. Cette heuristique ne conserve que les liens prédits conjointement dans les deux directions, qui correspondent intuitivement aux alignements les plus surs.

Les étiquettes morpho-syntaxiques pour les phrases sources en anglais sont prédites automatiquement en utilisant un modèle CRF linéaire standard entrainé sur le Penn Treebank. Les étiquettes sont ensuite transférées vers les phrases cibles en utilisant la méthode décrite dans la partie 2. Pour le français et l'espagnol, nous avons réextrait les contraintes de type avec nos propres outils. Pour l'allemand et le grec, nous avons directement utilisé les contraintes extraites par Li *et al.* (2012) [10].

Comme le montrent les statistiques présentées dans le tableau 1, `Wiktionary` fournit des informations pour un grand nombre de mots des corpus d'apprentissage. Cette information reste cependant fortement ambigüe, mais peut être efficacement complétée par les informations extraites des alignements. Pour l'allemand par exemple, considérer conjointement les deux sources d'information permet de réduire le nombre moyen d'étiquettes par mot de 4,6 à 1,1. Ces deux sources d'informations sont complémentaires : le dictionnaire permet de filtrer les étiquettes transférées (entre 10 et 15% des étiquettes ne sont pas transférées en raison de leur incompatibilité avec les contraintes de type) ; en même temps, les étiquettes transférées permettent de lever l'ambigüité des contraintes de type pour environ 50% des mots-occurrences alignés.

8. La quasi-totalité des ressources utilisées dans les travaux antérieurs ne sont pas ou plus distribuées, ce qui complique fortement toute comparaison directe.
9. Sauf pour le grec, pour lequel nous n'utilisons que le corpus EUROPARL.
10. Ces ressources sont disponibles à l'url `https://code.google.com/p/wikily-supervised-pos-tagger/`

Au final l'utilisation conjointe des contraintes et les règles de transfert permettent de créer un corpus d'apprentissage fortement désambigüisé, puisque la plupart des mots-occurrences du corpus d'apprentissage se retrouvent associés à une seule étiquette, et seule une petite partie des mots correspond à plus de 3 étiquettes.

	contraintes	français	grec	espagnol	allemand
% des mots-occurrences dans `Wiktionary`		91,4%	66,0%	87,7%	69,3%
nombre moyen d'étiquettes par mot-occurrence	*wiki*	2,5	5,0	2,8	4,6
nombre moyen d'étiquettes par mot-occurrence	*wiki*$^+$	1,7	1,6	1,6	1,6
nombre moyen d'étiquettes par mot-occurrence	*wiki*$^+$ & *transfert*	1,3	1,1	1,3	1,1
% de mots-occurrences avec une seule étiquette	*wiki*$^+$ & *transfert*	79,4%	88,1%	78,4%	89,5%
% de mots-occurrences avec une ou deux étiquettes	*wiki*$^+$ & *transfert*	90,8%	99,6%	96,6%	99,7%
% de mots-occurrences cibles alignés		71,1%	73,9%	74,0%	69,9%
% d'étiquettes transférées		85,9%	88,2%	85,8%	88,8%
% d'étiquettes transférées informatives		39,0%	50,6%	38,8%	53,7%

TABLE 1 – Statistiques sur l'ambigüité des étiquettes par mot-occurrence dans les corpus parallèles d'apprentissage après filtrage par le dictionnaire extrait de `Wiktionary` (*wiki*), puis lorsque l'on complète celui-ci avec des contraintes de types extraites des alignements (*wiki*$^+$) (section 2.3) et finalement en utilisant la méthode de transfert introduite à la section 2.4 (*transfert*). Le pourcentage d'étiquettes transférées correspond au pourcentage de liens d'alignements pour lesquels l'étiquette transférée est dans les contraintes de type ; le pourcentage d'étiquettes informatives correspond au pourcentage de liens d'alignements pour lesquels l'étiquette transférée est dans les contraintes de type, mais uniquement lorsque ces contraintes de types sont ambigües.

Notre approche est évaluée pour chaque langue considérée sur les ensembles de test des campagnes d'évaluation d'analyse en dépendances (Buchholz & Marsi, 2006; Nivre *et al.*, 2007) [11]. Ces corpus ont été constitués manuellement par des experts linguistes et contiennent plusieurs types d'annotations, dont des étiquettes morpho-syntaxiques fines qui sont transformées en leur équivalent dans le jeu d'étiquettes universelles en utilisant les règles de (Petrov *et al.*, 2012). La qualité des analyseurs entrainés est évaluée par leur taux d'erreur par occurrence sur le jeu de test.

4.2 Traits

Dans toutes nos expériences, nous utilisons des traits similaires à ceux qui sont généralement utilisés dans des tâches d'analyse morpho-syntaxique :

- Pour le mot courant ainsi pour que les deux mots précédents et les deux mots suivants :
  - **identité du mot** : mot en minuscules s'il apparait plus de 10 fois dans le corpus d'apprentissage ;
  - **suffixes** : les suffixes de 2 et 3 lettres s'ils apparaissent dans plus de 20 mot-types différents dans le corpus d'apprentissage ;
  - **classe** : la classe de ce mot [12] parmi 50 classes estimées sur le corpus d'apprentissage en utilisant MKCLS [13]. Les clusters de mots, appris de manière non supervisée, ont déjà étés utilisés comme traits pour améliorer les performances nombreuses de tâches de TAL (Koo *et al.*, 2008; Täckström *et al.*, 2012; Owoputi *et al.*, 2013; Täckström *et al.*, 2013) ;
- **majuscule** : deux traits binaires qui indiquent si le mot courant commence par une majuscule ou non ;
- **trait d'union** : deux traits binaires qui indiquent si le mot courant comporte un trait d'union ou non ;
- **type d'alphabet** : deux traits qui indiquent si le mot est écrit dans un alphabet grec ou latin ;
- **information de structure** : les étiquettes prédites pour les deux mots précédents, la conjonction de ces deux étiquettes, la conjonction de l'étiquette précédente et du mot précédent.

Ces caractéristiques sont semblables à celles sont qui sont utilisées dans (Täckström *et al.*, 2013; Li *et al.*, 2012), exceptées les informations de structure qui ne peuvent être facilement considérées dans un modèle de séquence linéaire comme les CRF.

11. Pour le français, le corpus de test est constitué des 2 000 premières phrases du French Treebank.
12. Les mots hors vocabulaire lors du test sont arbitrairement associés à la classe 1.
13. `http://code.google.com/p/giza-pp/`

4.3 Conditions expérimentales

Pour toutes les paires de langues considérées, un analyseur morpho-syntaxique est entrainé à partir des étiquettes ambigües. Le nombre d'itérations dans l'algorithme 2 est fixé à $T = 100\,000$, ce qui revient à dire que les paramètres de notre méthode sont estimés sur un sous-corpus de 100 000 phrases choisies aléatoirement dans le corpus d'apprentissage. Nos expériences préliminaires indiquent que le choix de ces phrases n'a que peu d'impact sur les performances obtenues. Il apparait également que considérer plus de phrases ne permet pas d'améliorer les performances.

Nous donnons également les résultats pour une réimplémentation du modèle CRF partiellement observé de (Täckström *et al.*, 2013) avec le même jeu de traits que les auteurs, en utilisant 30 itérations de R-Prop sur 100 000 phrases du corpus d'apprentissage et une régularisation ℓ_1 et ℓ_2 [14].

4.4 Résultats

Les résultats obtenus par notre méthode sont résumés dans le tableau 2. Les meilleurs scores de (Täckström *et al.*, 2013) et (Li *et al.*, 2012) pour les langues considérées y sont également inclus, même si ceux-ci ne sont pas directement comparables [15] puisque les différents modèles n'ont pas été entrainés exactement à partir des mêmes ressources (dictionnaires extraits de `Wiktionary`, corpus d'apprentissage, méthode d'alignement, etc).

Ce tableau montre que pour trois des quatre langues, notre méthode améliore sensiblement les résultats de l'état de l'art. L'utilisation de ressources de meilleure qualité, notamment pour les contraintes de type qui ont été extraites à partir d'une version plus récente de `Wiktionary` pour le français et l'espagnol, peut expliquer une partie des gains observés. Cependant, les résultats obtenus sur le grec, pour lequel nous utilisons les mêmes ressources que (Li *et al.*, 2012) et donc plus anciennes que celles utilisées dans Täckström *et al.* (2013), semblent indiquer qu'au moins une partie des améliorations est imputable à la méthode d'apprentissage introduite dans ce travail. De plus en entrainant sur nos ressources un modèle CRF partiellement observé, nous obtenons des résultats comparables à ceux de (Täckström *et al.*, 2013) pour leur modèle équivalent [16], à l'exception de l'allemand. Des expériences supplémentaires sont cependant nécessaires pour déterminer plus précisément les raisons de ces améliorations.

	français	grec	espagnol	allemand
Méthode proposée	**8,9%**	**8,3%**	**7,0%**	10,1%
CRF partiellement observé	13,2%	10,6%	14,0%	18,9%
meilleur score de (Täckström *et al.*, 2013)	11,6%	10,5%	10,9%	**9,5%**
meilleur score de (Li *et al.*, 2012)	—	20,8%	13,6%	14,2%
Inexactitude	5,9%	1,5%	3,4%	3,2%

TABLE 2 – Performances obtenues par notre méthode, un CRF partiellement observé similaire à $\hat{\mathcal{Y}}^{\text{CRF}}_{\text{wik.}}$+C+L dans (Täckström *et al.*, 2013) et les méthodes de l'état de l'art. L'inexactitude est le pourcentage de mots-occurrences dans le corpus de test pour lesquels la contrainte de type n'inclut pas l'étiquette de la référence et correspond à la meilleure performance que pourrait atteindre notre système.

4.5 Discussion

En première analyse, les résultats obtenus par les méthodes de transfert semblent encore très éloignés des performances des meilleurs étiqueteurs morpho-syntaxiques entrainés de manière supervisée. Ainsi, pour l'espagnol, un modèle CRF utilisant les mêmes traits que notre méthode (partie 4.2) appris sur le corpus d'entrainement des données CONLL, atteint un taux d'erreur de seulement 1,3% sur les données de test, contre 7,0% pour notre méthode. Il faut toutefois noter que l'évaluation des approches comme la nôtre comporte un fort biais. En effet, dans la majorité des travaux sur l'étiquetage

14. Dans un contexte de langue cible peu dotée en ressources il n'est pas possible d'utiliser un corpus de développement pour choisir les hyperparamètres des modèles. Comme dans (Täckström *et al.*, 2013), nous fixons arbitrairement les hyperparamètres du CRF partiellement observé à 1.

15. Il faut également noter que Täckström *et al.* (2013) et Li *et al.* (2012) proposent tous deux plusieurs méthodes et que seul le meilleur résultat obtenu sur le corpus de test (et non sur un corpus de validation) est présenté.

16. Sur le tableau 2, seul le meilleur modèle de (Täckström *et al.*, 2013) est indiqué. Le modèle CRF partiellement observé est leur meilleur modèle pour le grec et pour l'allemand.

morpho-syntaxique, l'évaluation est réalisée sur des corpus du même domaine que les corpus d'entrainement, comme pour le modèle CRF introduit ci-dessus. Les méthodes exploitant un transfert bilingue, en revanche, reposent sur des corpus d'apprentissage parallèles, qui peuvent être plus ou moins proches du corpus de test. Par ailleurs, les données de test utilisées exploitent une segmentation en mots qu'il n'est pas toujours aisé de reproduire à l'apprentissage et les conventions d'étiquetage ne sont pas nécessairement les mêmes que celles qui sont utilisées lors de l'apprentissage. Si le premier problème n'a qu'un impact limité sur les performances (il ne concerne que des mots isolés et n'a donc pas d'impact systématique) le second soulève un problème plus fondamental de notre approche ou, du moins, de son évaluation.

L'étiquetage d'un corpus repose sur des conventions qui peuvent varier d'une campagne d'annotation à une autre. Si ces conventions ne sont pas les mêmes pour les corpus de test et d'apprentissage, les prédictions seront entachées d'erreurs systématiques et l'estimation des performances sera biaisée. La situation est encore plus compliquée dans le cadre du transfert d'annotations dans lequel les méthodes utilisent généralement plusieurs sources de données (dans notre cas : `Wiktionary`, le corpus parallèle et le corpus de test) dont les étiquettes doivent toutes être mises en correspondance avec le jeu d'étiquettes universelles. Ce problème est exacerbé dans l'évaluation des méthodes de transfert, dans la mesure où les étiquettes du corpus d'apprentissage sont transférées à partir d'un corpus construit indépendamment de celui utilisé pour l'évaluation. Par exemple, dans le corpus français issu du French Treebank utilisé lors de notre évaluation, les nombres sont étiquetés soit comme des déterminants (DET), par exemple dans le fragment « Christian Blanc, 44 ans » ou « un prêt de 25 millions de dollars », soit comme des adjectifs (ADJ), comme dans « le Monde du 12 janvier » ou « à la page 23 ». Dans le Penn Treebank en revanche, sur lequel sont apprises les étiquettes de la langue source qui seront transférées sur la langue cible, les nombres sont systématiquement associés à l'étiquette NUM. Nous pensons que cette différence est davantage due à un choix de convention qu'à une réalité linguistique. De la même manière, dans le corpus de test pour l'espagnol, *poco* (peu) est majoritairement étiqueté comme un déterminant alors que dans le Penn Treebank, *few* est systématiquement étiqueté comme un adjectif et que `Wiktionary` identifie *poco* soit comme un pronom, un adjectif ou un nom.

Pour évaluer l'impact des différences de conventions d'annotation ainsi que de l'effet du changement de domaine il faudrait pouvoir entrainer un analyseur morpho-syntaxique de manière supervisée sur les données parallèles utilisées lors de l'apprentissage de notre méthode faiblement supervisée. Comme il n'existe pas, à notre connaissance, de données parallèles étiquetées morpho-syntaxiquement, nous avons créé un tel corpus de manière artificielle en étiquetant automatiquement les phrases cibles du corpus parallèle espagnol à l'aide d'un analyseur en catégories morpho-syntaxiques état de l'art [17]. Un CRF entrainé sur ces données obtient un taux d'erreur sur le corpus de test de $6{,}7\%$. Cette valeur, proche de celle obtenue par transfert, montre bien que la principale source d'erreur de notre méthode est liée aux différences de domaine et de convention d'annotation et non à des limites intrinsèques de la méthode d'apprentissage ou de transfert.

Il faut également considérer, dans l'analyse des performances obtenues, que les systèmes faiblement supervisés utilisant les contraintes de type sont fortement limités par l'exactitude de ces contraintes (cf. tableau 2). À titre d'exemple, pour l'espagnol, si l'on contraint le CRF appris de manière supervisée sur le corpus d'apprentissage de CoNLL à ne choisir que des étiquettes autorisées par les contraintes de type, sa performance chute de $1{,}3\%$ à $4{,}3\%$. L'exactitude des contraintes de type dépend elle aussi largement des conventions d'annotation. Pour le français, le taux élevé d'inexactitude s'explique principalement par un faible nombre de mots-types très fréquents (par exemple « au » ou « du ») dont les étiquettes morpho-syntaxiques diffèrent de manière systématique entre `Wiktionary` et le corpus de test.

L'évaluation de notre méthode, et plus généralement des méthodes faiblement supervisées, pose donc de nombreux problèmes méthodologiques, qui sont pourtant rarement discutés. L'interprétation des résultats obtenus doit être faite avec précaution, en particulier lorsqu'il s'agit de les comparer avec des méthodes supervisées.

5 Conclusion

Nous considérons dans ce travail le problème de l'apprentissage d'un analyseur morpho-syntaxique lorsque les étiquettes de supervision ne sont que partiellement connues, par exemple lorsque celles-ci sont automatiquement transférées à partir d'une langue source plus riche en annotations. En abordant ce problème sous l'angle de l'apprentissage ambigu, nous montrons qu'il est possible d'étendre un modèle à base d'historique capable d'apprendre dans un contexte faiblement supervisé. Pour trois des quatre langues considérées, notre méthode améliore sensiblement les résultats les plus récents. Enfin, nous discutons des difficultés et des limites que l'évaluation de telles méthodes pose. En particulier, les différences

17. Nous avons utilisé dans nos expériences FREELING (`http://nlp.lsi.upc.edu/freeling/`)

de convention entre différents corpus annotés peuvent largement biaiser les résultats. Il apparait au final, que la mise en œuvre et l'évaluation des méthodes de transfert nécessitent tout de même un effort conséquent et un minimum de connaissances des langues mises en jeu, ce qui conduit à en relativiser en quelque sorte l'intérêt.

Remerciements

Nous tenons à remercier nos relecteurs anonymes pour leurs très nombreux commentaires ainsi que Thomas Lavergne pour l'implémentation du CRF partiellement supervisé.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building a treebank for French. In A. ABEILLÉ, Ed., *Treebanks : Building and Using Parsed Corpora*. Dordrecht : Kluwer.

BANKO M. & MOORE R. C. (2004). Part of speech tagging in context. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA : Association for Computational Linguistics.

BLACK E., JELINEK F., LAFFERTY J., MAGERMAN D. M., MERCER R. & ROUKOS S. (1992). Towards history-based grammars : Using richer models for probabilistic parsing. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '91, p. 134–139, Stroudsburg, PA, USA : Association for Computational Linguistics.

BORDES A., USUNIER N. & WESTON J. (2010). Label ranking under ambiguous supervision for learning semantic correspondences. In *ICML*, p. 103–110.

BROSCHART J. (2009). Why Tongan does it differently : Categorial distinctions in a language without nouns and verbs. *Linguistic Typology*, **1**, 123–166.

BROWN P. F., DESOUZA P. V., MERCER R. L., PIETRA V. J. D. & LAI J. C. (1992). Class-based n-gram models of natural language. *Comput. Linguist.*, **18**(4), 467–479.

BUCHHOLZ S. & MARSI E. (2006). CoNLL-X Shared Task on Multilingual Dependency Parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, CoNLL-X '06, p. 149–164, Stroudsburg, PA, USA : Association for Computational Linguistics.

CHRISTODOULOPOULOS C., GOLDWATER S. & STEEDMAN M. (2010). Two decades of unsupervised POS induction : How far have we come ? In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, p. 575–584, Stroudsburg, PA, USA : Association for Computational Linguistics.

COLLINS M. (2003). Head-driven statistical models for natural language parsing. *Comput. Linguist.*, **29**(4), 589–637.

COUR T., SAPP B. & TASKAR B. (2011). Learning from partial labels. *Journal of Machine Learning Research*, **12**, 1501–1536.

DAS D. & PETROV S. (2011). Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies - Volume 1*, HLT '11, p. 600–609, Stroudsburg, PA, USA : Association for Computational Linguistics.

DAUMÉ, III H. & MARCU D. (2005). Learning as search optimization : Approximate large margin methods for structured prediction. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, p. 169–176, New York, NY, USA : ACM.

EVANS N. & LEVINSON S. C. (2009). The myth of language universals : Language diversity and its importance for cognitive science. *Behavioral and Brain Sciences*, **32**, 429–448.

GARRETTE D. & BALDRIDGE J. (2013). Learning a part-of-speech tagger from two hours of annotation. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 138–147, Atlanta, Georgia : Association for Computational Linguistics.

KAZAMA J. & TORISAWA K. (2007). A new perceptron algorithm for sequence labeling with non-local features. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, p. 315–324.

KOEHN P., HOANG H., BIRCH A., CALLISON-BURCH C., FEDERICO M., BERTOLDI N., COWAN B., SHEN W., MORAN C., ZENS R., DYER C., BOJAR O., CONSTANTIN A. & HERBST E. (2007). Moses : Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, p. 177–180, Stroudsburg, PA, USA : Association for Computational Linguistics.

KOO T., CARRERAS X. & COLLINS M. (2008). Simple semi-supervised dependency parsing. In *Proceedings of ACL-08 : HLT*, p. 595–603, Columbus, Ohio : Association for Computational Linguistics.

LI S., GRAÇA J. A. V. & TASKAR B. (2012). Wiki-ly supervised part-of-speech tagging. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, p. 1389–1398, Stroudsburg, PA, USA : Association for Computational Linguistics.

MANNING C. D. (2011). Part-of-speech tagging from 97% to 100% : Is it time for some linguistics ? In *Proceedings of the Computational Linguistics and Intelligent Text Processing, 12th International Conference, CICLing 2011*, p. 171–189 : Springer.

MERIALDO B. (1994). Tagging english text with a probabilistic model. *Comput. Linguist.*, **20**(2), 155–171.

NIVRE J., HALL J., KÜBLER S., MCDONALD R., NILSSON J., RIEDEL S. & YURET D. (2007). The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, p. 915–932, Prague, Czech Republic : Association for Computational Linguistics.

OWOPUTI O., O'CONNOR B., DYER C., GIMPEL K., SCHNEIDER N. & SMITH N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 380–390, Atlanta, Georgia : Association for Computational Linguistics.

PETROV S., DAS D. & MCDONALD R. (2012). A universal part-of-speech tagset. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, M. U. DOĞAN, B. MAEGAARD, J. MARIANI, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey : European Language Resources Association (ELRA).

RATINOV L. & ROTH D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, CoNLL '09, p. 147–155, Stroudsburg, PA, USA : Association for Computational Linguistics.

RESNIK P. & SMITH N. A. (2003). The web as a parallel corpus. *Computational Linguistics*, **29**(3), 349–380.

ROSS S. & BAGNELL D. (2010). Efficient reductions for imitation learning. In *AISTATS*, p. 661–668.

TÄCKSTRÖM O., MCDONALD R. & USZKOREIT J. (2012). Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 477–487, Stroudsburg, PA, USA : Association for Computational Linguistics.

TOUTANOVA K. & JOHNSON M. (2007). A Bayesian LDA-based model for semi-supervised part-of-speech tagging. In *NIPS*.

TSURUOKA Y., MIYAO Y. & KAZAMA J. (2011). Learning with lookahead : Can history-based models rival globally optimized models ? In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, p. 238–246, Portland, Oregon, USA : Association for Computational Linguistics.

TÄCKSTRÖM O., DAS D., PETROV S., MCDONALD R. & NIVRE J. (2013). Token and type constraints for cross-lingual part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, **1**, 1–12.

WANG M. & MANNING C. D. (2014). Cross-lingual projected expectation regularization for weakly supervised learning. *Transaction of the ACL*, **2**(1), 55–66.

WOLPERT D. H. (1992). Stacked generalization. *Neural Networks*, **5**.

YAROWSKY D., NGAI G. & WICENTOWSKI R. (2001). Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the First International Conference on Human Language Technology Research*, HLT '01, p. 1–8, Stroudsburg, PA, USA : Association for Computational Linguistics.

Construire un corpus monolingue annoté comparable
Expérience à partir d'un corpus annoté morpho-syntaxiquement

Nicolas Hernandez
LINA UMR 6241, Université de Nantes
nicolas.hernandez@univ-nantes.fr

Résumé. Motivé par la problématique de construction automatique d'un corpus annoté morpho-syntaxiquement distinct d'un corpus source, nous proposons une définition générale et opérationnelle de la relation de la comparabilité entre des corpus monolingues annotés. Nous proposons une mesure de la relation de comparabilité et une procédure de construction d'un corpus comparable annoté à partir d'un corpus annoté existant. Nous montrons que la mesure de la perplexité (théorie de l'information) est un moyen de sélectionner des phrases nouvelles pour construire un corpus comparable annoté grammaticalement.

Abstract. This work is motivated by the will of creating a new part-of-speech annotated corpus in French from an existing one. We propose a general and operational definition of the comparability relation between annotated monolingual corpora. We also propose a comparability measure and a procedure to build semi-automatically a comparable corpus from a source one. We study the use of the perplexity (information theory motivated measure) as a way to rank the sentences to select for building a comparable corpus. We show that the measure can play a role but that it is not sufficient.

Mots-clés : Corpus comparable, Corpus monolingue, Corpus annoté, Mesure de la comparabilité, Construction de corpus comparable, Analyse morpho-syntaxique, Auto-apprentissage, Perplexité.

Keywords: Comparable corpus, Monolingual corpus, Annotated corpus, Measuring comparability, Building comparable corpus, Part-of-Speech tagging, Self-learning, Perplexity.

1 Introduction

La question de construction de corpus comparables est souvent abordée dans le contexte applicatif d'extraction terminologique multilingue [1] (Bo et al., 2011). Cette question a aussi son importance dans un contexte monolingue où l'on peut vouloir construire un corpus avec des propriétés linguistiques similaires à un corpus source. En effet, il n'est pas rare pour un chercheur de souhaiter diffuser des corpus et des résultats d'analyse associés et ce, afin de permettre à ses pairs de vérifier ou de poursuivre ses expériences (Nielsen, 2011). Néanmoins, en pratique ce type d'ambition se trouve souvent compromis pour des questions de droit visant la protection de données personnelles ou le respect d'une licence d'exploitation associée aux données utilisées.

Dans ce travail, nous nous situons dans un contexte monolingue avec comme contraintes de ne pas pouvoir diffuser tout ou partie d'un corpus source mais d'avoir à disposition un corpus, que nous appellerons *corpus relais*, de taille importante et ne présentant pas les restrictions d'exploitation du corpus source. Dans ce cadre, nous explorons la possibilité de construire un corpus comparable au corpus source à partir d'extraits du corpus relais. Nous postulons que la détermination de la *relation de comparabilité* entre deux corpus est fonction du traitement d'analyse (e.g. segmentation en mots, étiquetage morpho-syntaxique, reconnaissance des entités nommées...) à laquelle on destine ces corpus. Plus précisément nous qualifierons de *comparables* [2] des corpus à partir desquels on peut construire des modélisations [3] pour un traitement donné, qui produisent des performances équivalentes lorsqu'elles sont évaluées sur un corpus tiers.

1. http://comparable.limsi.fr/bucc2013

2. Le qualificatif de «comparable» est généralement utilisé pour décrire un corpus dont les composantes (des sous-corpus) sont comparables. En ce qui nous concerne, nous utiliserons le terme «corpus comparable» pour désigner directement ces composantes. Par ailleurs, le qualificatif s'applique généralement à des composantes écrites dans des langues différentes, dont les textes ne sont pas en relation de traduction stricte. Dans ce travail, nous utilisons sciemment ce terme dans un contexte monolingue.

3. Ici le terme «modélisation» désigne aussi bien des règles construites manuellement qu'un modèle probabiliste.

Dans cet article, nous nous intéressons au problème de comparabilité en termes d'annotation morpho-syntaxiques. Nous supposons à disposition un corpus source annoté morpho-syntaxiquement. Nous nous interrogeons sur la possibilité de construire automatiquement un corpus comparable distinct à partir duquel on puisse entraîner un étiqueteur dont la performance ne sera statistiquement pas différente de celle d'un étiqueteur construit à partir du corpus source.

Ce travail poursuit les travaux de (Hernandez & Boudin, 2013) qui ont montré qu'un étiqueteur entraîné sur un corpus annoté automatiquement pouvait obtenir des performances statistiquement non différentes d'un étiqueteur entraîné sur un corpus annoté validé manuellement, et ce pourvu que la quantité de données d'entraînement fût suffisante. En d'autres termes, les auteurs ont montré que sous certaines conditions un corpus avec une annotation manuellement validée et un corpus automatiquement annoté pouvaient être comparables. Nous nous posons ici la question de savoir si l'observation tenue par (Hernandez & Boudin, 2013) repose seulement sur la quantité des données d'entraînement ou bien si une sélection éclairée des phrases d'entraînement pourrait conduire au même résultat.

Dans la section 2, nous proposons une définition générale et opérationnelle de la notion de comparabilité. Nous l'accompagnons d'une proposition d'une procédure générique visant la construction d'un corpus comparable. A la section 3, nous instancions notre réflexion sur la problématique de construction d'un corpus comparable annoté morpho-syntaxiquement en mettant notamment en avant la possibilité d'utiliser la mesure de perplexité sur les mots (théorie de l'information) comme moyen d'ordonner les phrases à sélectionner pour constituer un corpus comparable. Après avoir décrit notre cadre expérimental à la section 4, nous rapportons à la section 5 les résultats d'expériences de construction menées à partir de différents corpus relais. Enfin nous discutons notre approche par rapport à l'existant 6.

2 Corpus comparables monolingues

Nous posons qu'un corpus peut être vu comme un ensemble d'unités textuelles et que ses unités illustrent des phénomènes linguistiques de différentes natures (lexicales, syntaxiques, sémantiques, stylistiques, discursives...). Un traitement d'analyse sera sensible à un sous-ensemble défini de ces phénomènes. Par exemple, la prédiction d'une étiquette grammaticale pour un mot donné peut être fonction des éléments lexicaux et flexionnels des mots qui le précèdent. Le résultat du traitement sera fonction d'une part d'une modélisation des phénomènes, construite a priori par l'observation de ceux-ci au sein d'un corpus d'entraînement et d'autre part, de l'observation de la distribution de ces phénomènes dans le corpus nouvellement analysé. Pour la tâche d'étiquetage grammatical d'un mot, la sélection des trois derniers caractères du mot qui précède pourra constituer une observation ; sa probabilité d'occurrence pour prédire une certaine étiquette constituera une modélisation. La nature des unités textuelles manipulées est contrainte par le type d'information que le traitement d'analyse requiert en entrée. Pour l'étiquetage grammatical c'est généralement la phrase.

Soient trois corpus à disposition : un *corpus source* S pour lequel on veut construire un corpus comparable, un *corpus relais* R dont on souhaite extraire le corpus comparable $\tilde{R}$ et un *corpus de référence* Q sur lequel on peut évaluer et comparer des modélisations construites à partir du corpus source et d'extraits du corpus relais. Soit t un traitement d'analyse. Soient F_t l'ensemble des phénomènes discriminants pour la tâche t. Soient $o(F_t, \tilde{R})$ et $o(F_t, S)$, les observations que l'on peut faire de F_t (selon une même procédure) respectivement dans les corpus $\tilde{R}$ et S. Soient $m(o(F_t, \tilde{R}))$ et $m(o(F_t, S))$, des modélisations construites selon une même procédure et qui capturent la distribution des phénomènes discriminants pour la tâche t respectivement sur les corpus $\tilde{R}$ et S. Nous noterons $card_{\tilde{R}}(x)$ et $card_S(x)$ le nombre d'occurrences du phénomène linguistique x observés respectivement dans les corpus $\tilde{R}$ et S.

2.1 Propriétés d'un corpus en relation de comparabilité avec un autre corpus

Nous énonçons qu'un corpus $\tilde{R}$ est dit comparable à un corpus S (i.e. $\tilde{R}\, \mathcal{C}\, S$) si l'on peut observer les propriétés suivantes :

Propriété S et $\tilde{R}$ ne sont pas constitués des mêmes unités textuelles.

$$S \neq \tilde{R} \Leftrightarrow \forall u, \text{unité textuelle}, (u \in S \implies u \in \tilde{R}) \wedge (u \in \tilde{R} \implies u \in S) \quad (1)$$

Propriété Les phénomènes linguistiques discriminants pour la tâche t observés dans les unités de S sont aussi observés dans celles de $\tilde{R}$,

$$o(F_t, S) \subseteq o(F_t, \tilde{R}) \Leftrightarrow \forall x, (x \in o(F_t, S) \Rightarrow x \in o(F_t, \tilde{R})) \quad (2)$$

Propriété ... et ce, dans les mêmes proportions [4].

$$\forall x \in o(F_t, S), \forall y \in o(F_t, \tilde{R}), (x = y \Rightarrow card_S(x) = \lambda * card_{\tilde{R}}(y)) \text{ avec } \lambda > 0 \quad (3)$$

2.2 Mesure de la relation de comparabilité

En pratique, il n'est pas simple de mesurer ces propriétés. La raison principale vient du fait qu'il n'est pas aisé d'appréhender précisément un phénomène linguistique discriminant pour une tâche. En effet, un phénomène peut être porté par une ou plusieurs expressions textuelles distinctes. Celles-ci ne sont pas toujours simples à délimiter et peuvent participer à l'expression de plusieurs phénomènes dans un texte. C'est pour cette raison que nous proposons de comparer des corpus à travers la comparaison des résultats qu'obtiennent des systèmes entraînés respectivement sur chacun d'eux. Les systèmes définissent, de fait, un type d'information en entrée qui leur permettent d'observer les phénomènes opportuns.

La comparaison de systèmes est possible à travers un *test statistique de significativité* qui va permettre de mesurer la significativité statistique des différences entre deux ensembles de scores. Ce type de test retourne une probabilité $pvalue$ que l'on discute par rapport à des seuils pré-établis. Suivant les contextes applicatifs et les communautés scientifiques, les seuils α considérés sont 0,01, 0,05 et 0,1. Une valeur de probabilité s'interprète comme suit. Si elle est inférieure à un seuil de 0,05 par exemple, alors on peut affirmer avec moins de 5% de risques de se tromper (c'est-à-dire avec un niveau de confiance de 95%) que les scores sont significativement différents. De la même manière, si elle est supérieure à un seuil de 0,05, alors on peut affirmer que les scores ne sont pas significativement différents.

Pour obtenir les ensembles de scores, on découpe notre corpus de référence Q en n partitions et on évalue les systèmes sur chacune de ces partitions pour obtenir un score par partition et par système.

Ainsi, si les corpus $\tilde{R}$ et S sont comparables du point de vue du traitement t alors les modélisations $m(o(F_t, \tilde{R}))$ et $m(o(F_t, S))$, mises en oeuvre dans le traitement t pour analyser les n partitions Q_i d'un corpus tiers, donneront des résultats res_i ne présentant pas de différences statistiquement significatives.

$$\underset{i \in [1,n]}{pvalue} \left(res_i(t(m(o(F_t, \tilde{R})), Q_i)), res_i(t(m(o(F_t, S)), Q_i)) \right) > \alpha \quad (4)$$

2.3 Procédure de construction d'un corpus comparable

On souhaite construire un corpus $\tilde{R}$ à partir d'extraits d'un corpus R qui permette d'entraîner un système pour un traitement t dont les performances sont comparables à celles obtenues avec un système entraîné sur un corpus source S. Donné un corpus relais très grand, une possibilité est d'explorer toutes les combinaisons d'unités textuelles (e.g. les phrases) possibles jusqu'à en trouver une qui vérifie notre souhait. Cette approche n'est raisonnablement pas envisageable essentiellement pour des raisons de combinatoire (proportionnelle au nombre de phrases dans nos corpus) et de temps de calcul (construction des modélisations). Il est nécessaire d'opter pour un moyen simple capable de sélectionner les unités de R en fonction de S mais aussi selon une certaine sensibilité à la tâche t.

On définit le problème de construction d'un corpus comparable comme un *problème d'ordonnancement* des unités textuelles et *de recherche du nombre minimal* d'unités à atteindre pour constituer un ensemble qui satisfait aux mieux les propriétés énoncées à la section 2.1. Dans un premier temps, on construit à partir du corpus source S une modélisation pour la tâche t qui permet d'obtenir des scores sur les différentes partitions du corpus de référence Q.

Dans un second temps, on débute un mécanisme itératif qui vise la construction incrémentale du corpus comparable à l'aide d'un mécanisme de sélection prédéfini des unités textuelles du corpus relais R. Chaque itération donne lieu à un corpus comparable candidat $\tilde{R}_j$. Pour chacun d'eux, on construit une modélisation que l'on évalue sur les partitions du corpus de référence. On compare ensuite chaque modélisation avec celle construite sur le corpus source. On reproduit la procédure jusqu'à ce que l'on constate ne pas observer de différences statistiquement significatives entre les modélisations ou bien qu'il n'y ait plus d'unités à sélectionner.

4. Au sujet des propriétés 2 et 3, nous formulons la relation de comparabilité comme n'étant pas nécessairement une relation symétrique. En effet, un corpus reconnu comparable à un autre peut intégrer des phénomènes relatifs à une tâche non observés dans un corpus source. Le seul problème qu'il pourrait y avoir à cela est le fait de ne pas retrouver ces phénomènes dans le corpus de référence. Il en découle une difficulté à évaluer et à interpréter leurs rôles dans les résultats du système et dans les annotations produites.

Le procédé de sélection des unités textuelles du corpus relais est tel qu'à terme, il maximise la présence des phénomènes relatifs à la tâche t dans les proportions telles que celles observées dans le corpus S. En d'autres termes, ce procédé affecte un score à chaque unité qui permet de les prioriser entre elles.

Si le corpus relais n'est pas annoté, celui-ci peut l'être en appliquant le système construit à partir du corpus source.

3 Construction d'un corpus comparable annoté morpho-syntaxiquement

Un *modèle de langue probabiliste* constitue une représentation d'un corpus pour laquelle la théorie de l'information offre des moyens de comparaison peu coûteux. Entre autres, elle permet d'évaluer des modèles entre eux ou bien d'évaluer un modèle sur sa capacité à «reconnaître» un texte n'ayant pas participé à sa construction. Un modèle de langue est une fonction probabiliste p qui informe sur la probabilité d'occurrence d'une séquence de mots ($p(W) = p(w_1, w_2, ..., w_n)$) ou sur la probabilité de sortie d'un mot pour un historique de mots donné [5] ($p(w_n|w_1, w_2, ..., w_{n-1})$).

Dans le contexte de l'étiquetage morphosyntaxique, les probabilités de séquences lexicales sont connues pour être discriminantes (Toutanova et al., 2003). Par conséquent, nous posons l'hypothèse que si des modèles de langues estimés sur des corpus différents sont de qualité comparables alors les modélisations d'étiquetage morpho-syntaxique construites de façon similaire sur ces différents corpus sont de performances équivalentes.

Un moyen pour évaluer la qualité d'un modèle de langue est de mesurer la capacité du modèle à prédire un texte inconnu. Pour ce faire on peut utiliser la mesure de l'*entropie croisée* ou celle de la *perplexité* (mesure qui découle de la première et qui est communément employée en reconnaissance de la parole).

3.1 La perplexité comme critère d'ordonnancement

L'*entropie croisée* correspond au nombre moyen de bits requis pour encoder chacun des mots du texte inconnu. La *perplexité* peut être interprétée comme la capacité d'un modèle de langue à prédire le prochain mot donné son historique (les mots qui précédent) dans un texte inconnu (n'ayant pas servi pour la construction du modèle de langue évalué). Quand la distribution est uniforme, elle peut être interprétée comme un degré de ramification et indiquer le nombre de choix possibles pour le prochain mot. On peut aussi voir ce nombre de choix comme un degré de surprise.

Si l'on pose q comme étant la distribution empirique observée sur un texte inconnu (i.e. $q(w_i) = n_i/N$ pour le i^e mot qui apparaît n_i fois dans le texte inconnu de taille N), alors on peut définir l'entropie croisée $H(q, p)$ par la formule (5).

$$H(q,p) = -\sum_j q(w_j) \log_2 p(w_j) \approx -\frac{1}{N} \log_2 p(w_j) \quad (5)$$

avec j représentant le j^e mot dans les données de test
et avec N le nombre de mots dans les données de test

La perplexité est alors définie par la formule (6).

$$PP(W) = 2^{H(W)} \quad (6)$$

Une faible valeur d'entropie croisée pour des données inconnues indique que la distribution observée sur les données inconnues est *proche* de celle observée sur le modèle de langue. De la même manière, une valeur élevée de perplexité indique une mauvaise correspondance entre les données ayant servi à construire le modèle de langue et celles testées.

Pour des facilités d'interprétation nous choisissons d'utiliser la perplexité.

3.2 Construction d'un corpus comparable pour la tâche d'étiquetage morpho-syntaxique

Nous instancions la procédure décrite à la section 2.3 comme suit. Dans un premier temps, nous entraînons un étiqueteur sur l'ensemble d'un corpus source dont l'annotation a été validée manuellement.

5. les mots qui le précèdent

Puis nous utilisons ce système construit pour annoter automatiquement le corpus relais.

A partir de là, nous initions le processus de sélection de énoncés du corpus relais que nous poursuivons jusqu'à l'obtention du corpus comparable désiré ou l'appauvrissement total du corpus relais. Pour ce faire, nous estimons un modèle de langue sur le corpus source et nous calculons la perplexité du modèle sur chaque phrase du corpus relais. Les scores obtenus nous permettent d'ordonner les phrases entre elles selon une perplexité croissante. La phase de construction effective des corpus comparables candidats revient à sélectionner et à ajouter incrémentalement les n premières phrases du corpus relais non encore sélectionnées.

A partir de chaque paquet de phrases construit, nous entraînons un étiqueteur que nous évaluons sur un corpus de référence dont nous comparons les résultats avec un système construit directement à partir du corpus source.

Dans les expériences que nous rapportons ci-après nous ne stoppons pas la sélection de nouveaux énoncés quand nous estimons les systèmes comparés non différents. Nous rapportons des observations à différents étapes de la procédure itérative afin d'observer plus finement les courbes de comparabilité entre les corpus.

Nous utilisons les mesures classiques d'évaluation pour la tâche d'étiquetage morpho-syntaxique à savoir : la précision sur les mots (nombre de mots correctement étiquetés sur le nombre de mots total), la précision sur les phrases (nombre de phrases dans lesquelles tous les mots ont été correctement étiquetés par rapport au nombre de phrases total) et la précision sur les mots inconnus (nombre de mots inconnus correctement étiquetés sur le nombre de mots inconnus total [6]). Dans cet article, nous ne rapportons que la précision sur les mots.

Pour comparer les systèmes entre eux (ceux construits à partir d'extraits d'un corpus relais et celui construit sur le corpus source), nous utilisons un *test de Student* comme test de significativité. Ce test mesure si il y a une différence statistiquement significative entre les scores de précision obtenus par les différents systèmes. Il suppose une adéquation de la distribution des échantillons à la loi normale.

En pratique, ne possédant pas de corpus tiers de référence pour permettre l'évaluation des modélisations que nous construisons, nous utilisons le corpus source comme corpus de référence. Un système est construit sur l'ensemble du corpus source pour annoter le corpus relais. Avec $n = 10$, n systèmes sont construits à partir de $n - 1$ partitions du corpus source pour donner un score de précision sur la n^e partition (cf. la section 5.1). Tous les étiqueteurs entraînés à partir des extraits des différents corpus relais donnent un score de précision sur chacune de ces 10 partitions. Chaque système est alors décrit par cet ensemble de scores. Le test de Student compare alors les ensembles de scores décrivant deux systèmes pour déterminer si ils sont différents l'un de l'autre de manière significative.

4 Cadre expérimental

Dans cette section, nous présentons brièvement le corpus source et les corpus relais utilisé. Comme énoncé précédemment, le corpus source nous sert à la fois pour annoter les corpus relais et pour évaluer les systèmes construits à partir des corpus relais. Nous précisons aussi les implémentations utilisées pour segmenter les corpus en mots, les étiqueter morpho-syntaxiquement et manipuler les modèles de langue. Enfin, nous donnons quelques caractéristiques quantitatives en termes de taille du vocabulaire, nombre de mots et de phrases pour décrire les corpus.

4.1 Données

Le corpus arboré de Paris 7 (P7T) (Abeillé et al., 2003; Abeillé & Barrier, 2004), alias le *French Treebank* [7], se compose d'articles journalistiques issus du journal Le Monde couvrant la période 1989 à 1993. Ce corpus offre une analyse multi-niveaux (lexicale, morphologique et syntaxique) Sa licence propriétaire autorise une utilisation à des fins de recherche.

Wikinews figure parmi les projets de la Wikimedia Foundation [8]. Il s'agit d'un recueil de dépêches et de reportages d'actualité écrit par ses utilisateurs. La version francophone de Janvier 2013 compte plus de 28 000 articles d'actualité et couvre une période s'étalant de Janvier 2005 à nos jours. Les textes sont exploitables sous licence [9] Creative Commons Attribution 2.5 (CC-BY 2.5) (les textes antérieures à Septembre 2005 sont dans le domaine public) qui permet à l'util-

6. Calculé à partir des mots n'apparaissant pas dans l'ensemble d'entraînement
7. http://www.llf.cnrs.fr/Gens/Abeille/French-Treebank-fr.php
8. http://wikimediafoundation.org
9. http://dumps.wikimedia.org/legal.html

isateur d'utiliser, de modifier et de diffuser la ressource et ses modifications comme il le souhaite moyennant l'obligation d'en citer l'auteur.

L'Est républicain est un corpus de type journalistique mis à disposition par le CNRTL [10], et composé d'articles du quotidien régional du même nom. La version 0.3 met à disposition les éditions intégrales des années 1999, 2002 et 2003 exploitable sous license Creative Commons (CC BY-NC-SA 2.0 FR).

Europarl Ce corpus est constitué de textes multilingues alignés issus des actes du Parlement Européen [11] préparés par (Koehn, 2005) pour l'entraînement de systèmes statistiques de traduction. La section en français de la version 7 (mai 2012) couvre une période s'étalant de 1996 à 2011. Ces textes sont libres de reproduction [12].

Le corpus P7T sert de corpus source. Les corpus Wikinews, Est Républicain et Europarl servent de corpus relais. Les corpus Wikinews et Est Républicain présentent la particularité d'être du genre journalistique à l'instar du corpus source.

4.2 Segmentation des mots

Afin de comparer des étiqueteurs, il est important que ceux-ci aient été entraînés sur des textes segmentés en mots de la même manière. La question de la segmentation se pose surtout sur le traitement des mots composés. Le corpus P7T, qui constitue notre corpus source, fait reposer ses annotations sur une segmentation en mots composés. Afin de permettre à un système automatique de reproduire au plus près la segmentation du P7T nous avons réalisés un certain nombre d'adaptations (Hernandez & Boudin, 2013) pour ne considérer comme «mots» que les unités graphiques ne contenant pas d'espace. La segmentation de Wikinews et d'Europarl repose sur l'utilisation d'un même segmenteur [13] qui met en oeuvre ce principe. Nous l'avons étendu pour traiter l'Est Républicain qui contient davantage d'entités spécifiques.

La table 1 rapporte la taille du vocabulaire, le nombre de mots et le nombre de phrases pour chacun des corpus. La table 2, quant à elle, rapporte le taux de recoupement entre les différents vocabulaires des différents corpus ($\frac{|V_1 \cap V_2|}{|V_1 \cup V_2|}$).

	P7T	**Wikinews**	**Est Républicain**	**Europarl**
Taille du vocabulaire	34 677	75 175	382 342	129 093
Nombre de mots	629 788	2 535 396	36 478 209	61 396 216
Nombre de phrases	23 539	87 461	1 947 360	1 967 951

TABLE 1 – Taille du vocabulaire, nombre de mots et de phrases des corpus P7T, Wikinews, Est Républicain et Europarl.

	Wikinews (75 175)	**Est Républicain** (382 342)	**Europarl** (129 093)
P7T (34 677)	.27 (23 093 / 86 749)	.08 (29 831 / 387 188)	.20 (27 329 / 136 433)
Wikinews		.13 (52 068 / 405 439)	.30 (46 628 / 157 622)
Est Républicain			.17 (75 213 / 436 213)

TABLE 2 – Taux de vocabulaires en commun entre les corpus P7T, Wikinews, Est Républicain et Europarl.

4.3 Jeu d'étiquettes morpho-syntaxiques et étiqueteur

Le jeu de catégories morpho-syntaxiques que nous utilisons est celui mis ou point par (Crabbé & Candito, 2008), contenant 28 catégories qui combinent différentes valeurs de traits morpho- syntaxiques du P7T (désigné si après par P7T+). Outre le fait que ce jeu soit plus complet que les catégories du P7T, qui elles sont au nombre de 13, les auteurs montrent que les performances d'un étiqueteur entraîné sur de telles annotations sont meilleures.

En ce qui concerne l'étiqueteur morpho-syntaxique que nous avons utilisé pour nos expériences, il s'agit de la version 3.1.3 du *Stanford POS Tagger* (Toutanova et al., 2003). Ce système utilise un modèle par maximum d'entropie, et peut

10. http://www.cnrtl.fr/corpus/estrepublicain
11. http://www.statmt.org/europarl
12. «Except where otherwise indicated, reproduction is authorised, provided that the source is acknowledged.» http://www.europarl.europa.eu/guide/publisher/default_en.htm
13. https://github.com/boudinfl/kea

atteindre des performances au niveau de l'état-de-l'art en français (Boudin & Hernandez, 2012; Hernandez & Boudin, 2013). Nous utilisons un ensemble standard [14] de traits bidirectionnels sur les mots et les étiquettes.

4.4 Estimation de modèles de langue et calcul de la perplexité

Les modèles de langue que nous estimons sont d'ordre 5 (historique de 4 mots) et sont construits en utilisant la technique de lissage communément utilisée de (Kneser & Ney, 1995). En pratique, nous utilisons la bibliothèque *berkeleylm* [15] (Pauls & Klein, 2011). Pour calculer la perplexité nous utilisons la bibliothèque *kylm* [16].

5 Expériences

Dans les sections qui suivent nous rapportons les résultats de systèmes d'étiquetage entraînés à partir de différents corpus relais. Nous les évaluons sur les données du P7T+.

5.1 Performance d'un étiqueteur état-de-l'art

A titre de comparaison, la table 3 rapporte les résultats d'un système état de l'art (à savoir le *Stanford POS Tagger*) évalué par validation croisée en 10 strates sur le P7T+. Ces résultats peuvent être interprétés comme la performance maximale que peut obtenir un système lorsqu'il est entraîné sur des données qui ont été manuellement validées. La précision moyenne est de 96,93% sur les tokens. Nous renvoyons à (Hernandez & Boudin, 2013) pour une évaluation plus exhaustive en terme de précision sur les phrases et en tenant compte des mots inconnus.

	Précision	**Min. - Max.**	**Écart type**
Tokens	96,93	96,55 - 97,28	0,219

TABLE 3 – Scores de précision sur les tokens du *Stanford POS tagger* calculés à partir du P7T+ en validation croisée en 10 strates. Le minimum, le maximum et l'écart type des scores calculés sur les 10 strates sont également reportés.

5.2 Score de précision et test de significativité selon les corpus relais

Dans cette section nous rapportons pour différents corpus les scores de précision et de significativité (via le test de Student aussi appelé t-test) obtenus à partir de modélisations fondées sur un nombre croissant de phrases ordonnées soit aléatoirement [17] soit sur le score de perplexité vis-à-vis du corpus source.

Pour vérifier l'adéquation à la normalité de la distribution des échantillons et utiliser le test de Student, nous avons utilisé le *test de Shapiro-Wilk*. La taille de nos échantillons correspond au nombre de partitions de test, à savoir 10. Pour la grande majorité des échantillons (suffisamment pour soutenir les observations ci-dessous), l'hypothèse d'adéquation n'a pas été rejetée avec un seuil de 50 %.

Les figures 1, 2 et 3 rapportent les scores observés pour les corpus relais Wikinews, Est Républicain et Europarl. En abscisse de chaque figure, on note le nombre de phrases considéré pour la construction d'une modélisation. En ordonnée sur la gauche se trouve une échelle de scores du test de Student variants de 0 à 1 et en ordonnée sur la droite se trouve une échelle de scores de précision centrée sur les scores les plus élevés variants de 92% à 98% environ. Cette double échelle en ordonnée permet de confronter ces deux types de scores. Les courbes en pointillées (bleues et avec points ronds) représentent les scores de précision tandis que les courbes en ligne pleine (rouges et avec points carrés) correspondent aux scores du test de Student. Il y a pour ces trois figures deux courbes en pointillée et deux pleines. Celles marquées d'un point vide concernent les scores du test de Student et de précision obtenus à partir de modélisations fondées sur un nombre croissant de phrases ordonnées aléatoirement (ordre original). Celles marquées d'un point plein concernent les

14. Nous avons utilisé la macro `generic,naacl2003unknowns` décrite dans (Toutanova et al., 2003).
15. `https://code.google.com/p/berkeleylm`
16. `http://www.phontron.com/kylm/`
17. L'ordre original n'étant pas trié est considéré comme étant sans a priori.

scores du test de Student et de précision obtenus à partir de modélisations fondées sur un nombre croissant de phrases ordonnées sur le score de perplexité vis-à-vis du corpus source.

Dans la table 4, nous rapportons divers facteurs de nombres de phrases calculés entre différents corpus. Nous comparons les corpus obtenus à la valeur maximale du test de Student (pour notre échantillonnage) avec les corpus relais dont ils sont issus et le corpus source. Nous comparons aussi la différence du nombre de phrases entre les corpus triés et non triés.

Sur les trois figures 1, 2 et 3, on peut voir que la précision de tous les systèmes augmente avec la quantité de phrases. Les élévations observées sur les courbes du test de Student correspondent à des scores où le test considère qu'il n'y a pas de différences statistiquement significatives entre les systèmes entraînés sur les corpus relais et celui sur un corpus source. Il est à noter que les sommets de ces courbes sont sur la base de notre échantillonnage. Ils ne correspondent pas aux maximums absolus (probablement voisins).

Concernant le corpus relais Wikinews (figure 1), nous observons que le score de précision est meilleur pour le corpus relais sans ordre a priori que sur le corpus relais ordonné sur le score de la perplexité. La différence entre ces scores s'amenuise à mesure que l'on considère plus de phrases dans les corpus et finit par se confondre à partir d'une certaine quantité de phrases. Au sujet de la significativité des résultats, sur la base de notre échantillonnage, nous ne pouvons différencier les deux corpus candidats au maximum du test de Student. La courbe traduit une légère préférence pour le corpus ordonnée selon la perplexité mais la différence est minime.

Concernant le corpus relais Est Républicain (figure 2), les scores sont tout autre. Au sujet de la précision des systèmes, malgré une convergence qui se précise avec le nombre de phrases considéré, on observe qu'un système, construit en sélectionnant ses instances d'entraînement selon un score de perplexité avec le corpus source, obtient une précision plus haute qu'avec un système entraîné avec des phrases sans ordre a priori. Au sujet de la significativité, on constate qu'un système qui se fonde sur un ordre des phrases découlant d'un score de perplexité croissant requiert moins de phrases qu'un système qui ne se fonde pas sur un ordre a priori des phrases (1,85 fois la taille du corpus source pour le corpus trié contre 3,48 pour celui non trié). L'une des différences fondamentales avec le corpus relais Wikinews est que le corpus relais Est Républicain est beaucoup plus grand.

Cette expérience montre que le filtrage des phrases selon un score de perplexité qu'un corpus source peut avoir sur elles constitue un critère effectif de sélection.

Les scores observés à la figure 3 (corpus relais Europarl) corroborent globalement ceux observés à la figure 2. On note qu'il faut des quantités de données beaucoup plus importantes pour observer des résultats. On attribue cette caractéristique au fait que ce corpus relais n'est pas du même genre que les corpus source et relais Wikinews et Est Républicain. Dans cette expérience, on constate encore qu'avec des phrases sélectionnées selon un score de perplexité observé par un corpus source, une quantité moins importante est requise avec celles-ci pour obtenir plus rapidement des scores de précision avec une différence statistiquement non-significative. De même la précision sur mots est globalement toujours plus élevée.

	Wikinews		**Est Républicain**		**Europarl**		**Fusion**
Ordre des phrases dans corpus relais	aléa	trié	aléa	trié	aléa	trié	trié
# de phrases *au maximum du t-test*	76 524	76 524	82 000	43 730	349 844	262 383	43 730
Facteur *diviseur* / corpus relais	1.14	1.14	23.75	44.53	7.50	5.62	91.53
Facteur *multiplicateur* / corpus source	3.25	3.25	3.48	1.85	14.86	11.14	1.85
$\neq$ *entre # de phrases* aléa et trié	1		1.87		1.33		

TABLE 4 – Facteurs multiplicateur et diviseur calculés sur le nombre de phrases respectivement par rapport au corpus source et aux corpus relais, pour la valeur maximale du test de Student. Sont comparés les facteurs lorsque les phrases sont ordonnées aléatoirement ou selon la perplexité. Est donné le rapport entre le nombre de phrases non triées et triées pour le maximal du test de Student. Est aussi indiqué le nombre de phrases de l'échantillon pour lequel le score maximal du test de Student a été atteint.

5.3 Significativité obtenue à partir d'un corpus relais fusionné et ordonné

La figure 4 rapporte les scores de significativité et de précision obtenus à partir de systèmes construits sur un corpus relais résultant de la fusion des corpus relais Wikinews, Est Républicain et Europarl. Les courbes du corpus relais ordonné Est Républicain sont rappelées car les scores sont très similaires à ceux obtenus avec la fusion des corpus.

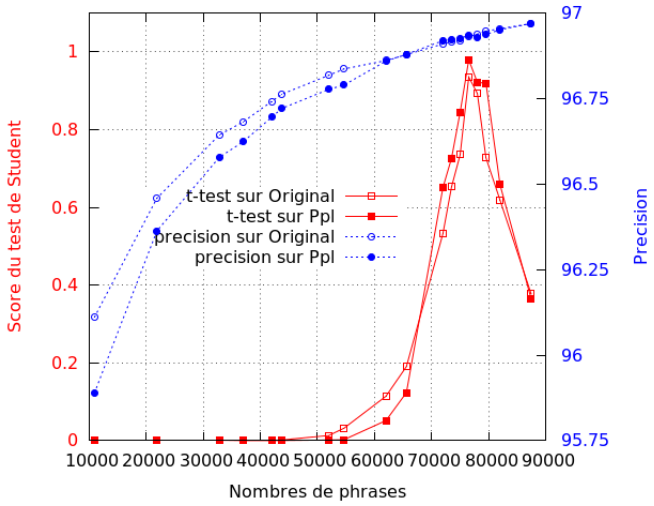


FIGURE 1 – Corpus Wikinews : Scores de précision et de t-test obtenus à partir de modélisations fondées sur un nombre croissant de phrases ordonnées soit aléatoirement soit sur le score de perplexité vis-à-vis du corpus source.

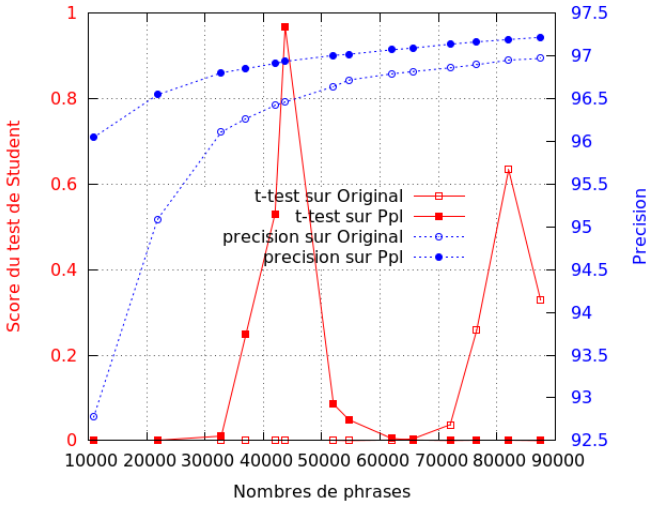


FIGURE 2 – Corpus Est Républicain : Scores de précision et de t-test obtenus à partir de modélisations fondées sur un nombre croissant de phrases ordonnées soit aléatoirement soit sur le score de perplexité vis-à-vis du corpus source.

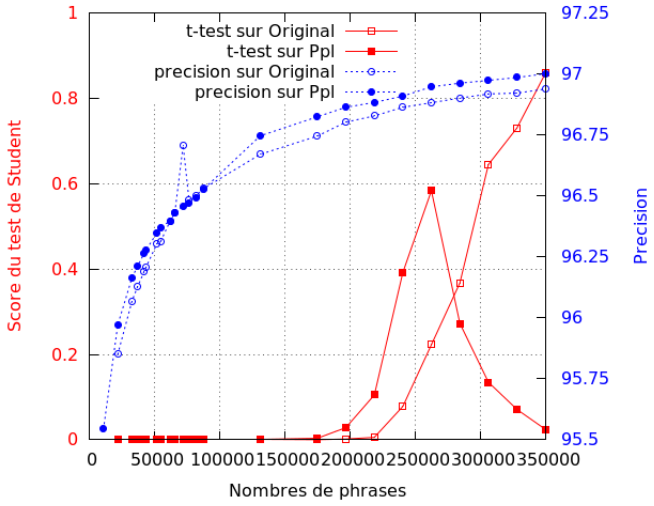


FIGURE 3 – Corpus Europarl : Scores de précision et de t-test obtenus à partir de modélisations fondées sur un nombre croissant de phrases ordonnées soit aléatoirement soit sur le score de perplexité vis-à-vis du corpus source.

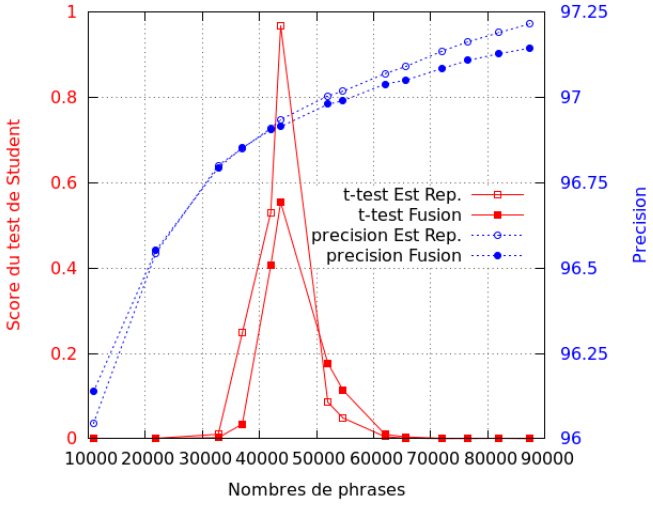


FIGURE 4 – Corpus fusionnés : Scores de précision et de t-test obtenus à partir de modélisations fondées sur un nombre croissant de phrases ordonnées sur le score de perplexité vis-à-vis du corpus source. Sont rappelés les scores du corpus Est Républicain.

Nous notons que le système construit sur le corpus fusionné requiert globalement le même nombre de phrases que celui construit à partir du meilleur corpus composant le corpus fusionné. Nous notons néanmoins que même si les deux courbes sont proches (courbe rouge avec carrés remplis et courbe rouge avec carrés vides), le système construit seulement à partir de l'Est Républicain obtient des scores de précision et de Student plus importants pour un échantillon donné.

Nous attribuons cela à une limitation de notre approche pour la construction des corpus comparables. En effet en l'état, la sélection des phrases à considérer se fait seulement sur la base du corpus source et de la phrase analysée. Elle ne prend pas en compte les phrases déjà sélectionnées et celles potentiellement sélectionnables. Ainsi, pour une même quantité de phrases, nous pensons que, dans le corpus résultant de la fusion, il y a plus de phrases similaires à celles du corpus source mais, proportionnellement, probablement moins de phrases qui couvrent l'étendue des phénomènes du corpus source par rapport au corpus construit à partir de l'Est Républicain.

Pour vérifier cette hypothèse nous calculons la perplexité de modèles de langue construits à partir de ces corpus sur ces différents corpus tour à tour. Outre le corpus source, nous considérons le corpus résultant de la fusion et celui de l'Est Républicain, tous deux au plus haut du score du test de Student. La table 5 rapporte ces résultats. On peut lire que le modèle de langue construit sur le corpus source P7T «reconnaît» plus facilement le corpus extrait de la fusion que celui issu de l'Est Républicain (perplexité de 267 contre une perplexité de 899). On en déduit que l'Est Républicain contient plus de cas inconnus du P7T que le corpus Fusion. On note aussi que le modèle de langue construit à partir du corpus issu de la fusion reconnaît moins bien le P7T que celui-ci ne le reconnaît (perplexité de 417 contre une perplexité de 267). Ces observations confirment notre hypothèse de différences de représentativité entre ces corpus. Le corpus Fusion semble contenir moins de diversité que le corpus P7T. A titre indicatif, nous donnons les perplexités des modèles de langue testés sur les corpus ayant servis à leur construction.

Modèle de langue \ Corpus testé	**P7T**	**Est Républicain**	**Fusion**
P7T	16	899	267
Est Républicain	553	18	228
Fusion	417	284	15

TABLE 5 – Scores de perplexité réciproque entre le corpus source P7T et les corpus comparables candidats extraits des corpus relais ordonné fusion et Est Républicain au score du test de Student le plus haut.

6 État de l'art

L'anonymisation est un moyen de transformation d'un corpus source qui dénature principalement les entités nommées et qui malgré cette perte d'information peut permettre la diffusion du dérivé et son exploitation dans des tâches autres que de la reconnaissance d'entités nommées (Medlock, 2006). Nous nous situons dans un contexte où nous ne pouvons réutiliser tout ou partie d'un corpus source.

(McEnery & Xiao, 2007) définit la notion de corpus comparable dans le cadre d'études comparatives inter-lingues. Selon les auteurs, un corpus comparable se définit comme un corpus dont les composantes ont été collectées en utilisant la même base d'échantillonnage, le même équilibre et la même représentativité par exemple la même proportion de textes de même genres dans les mêmes domaines collectés à la même période dans différentes langues. Nous avons testé différents corpus relais que nous avons sélectionnés en raison de leur taille et de leur licence d'exploitation. Nous sommes néanmoins d'avis que la sélection de textes de même registre et de la même période temporelle favorise l'obtention de résultats équivalents. Nous pensons néanmoins que la contrainte sur les langues peut être levée pour offrir une définition plus large.

L'approche classique que l'on retrouve pour construire des corpus comparables monolingues consiste à utiliser des moteurs de recherche et à utiliser les documents du corpus source comme requêtes (Wang & Callison-Burch, 2011; Bo et al., 2011). Le supposé théorique sous-jacent est qu'un corpus comparable à un autre partage un contenu informationnel (thématique) en commun. Nous pensons qu'il s'agit d'une restriction limitative de la définition de corpus comparable.

Dans le contexte de la tâche d'extraction terminologique bilingue, (Déjean & Éric Gaussier, 2002; Bo et al., 2011) définissent une mesure de comparabilité fondée sur le lexique partagé par les corpus comparés. A nouveau, cette définition illustre une importance prépondérante à l'information lexicale comme base à la comparabilité. Elle est néanmoins pertinente dans le cadre de cette tâche. Comme ces auteurs nous pensons qu'une définition de la notion de comparabilité est en lien

avec une application. Par notre approche, nous souhaitons néanmoins ne pas nous attacher à l'observation privilégiée de certains traits linguistiques. Notre définition se veut plus générale et non spécifique à une application particulière.

Notre approche pour construire un corpus annoté comparable est similaire à l'approche par apprentissage semi-supervisé de type auto-apprentissage (*self-training*). Ce type d'approche vise à étendre le taille d'un corpus annoté en ajoutant aux données d'entraînement des données nouvelles annotées par un système entraîné sur les données annotées initialement disponibles. Cette approche montre des résultats intéressants dans le contexte de l'adaptation de systèmes existants à de nouveaux domaines et lorsque les données d'entraînement ne sont seulement disponibles qu'en petite quantité (Rehbein, 2011). Pour ce qui nous concerne, les données du P7T sont estimées en quantités suffisantes pour pouvoir soutenir l'entraînement de systèmes statistiques type étiqueteur et analyseur syntaxique (Crabbé & Candito, 2008; Boudin & Hernandez, 2012). Par ailleurs, nous n'ajoutons pas les nouvelles données annotées au corpus initial mais les considérons pleinement comme de nouvelles données annotées à part entière.

Dans le contexte d'auto-apprentissage, on retrouve l'idée de favoriser les phrases les plus similaires aux données d'entraînement pour enrichir le corpus. L'objectif est de minimiser l'ajout de bruit dans les données. La mesure de perplexité (sur les étiquettes) a ainsi été utilisée par (Rehbein, 2011) et (Søgaard, 2011) pour construire des analyseurs syntaxiques. Nos résultats corroborent les leurs lorsqu'ils notent que la perplexité joue un rôle actif positif dans la sélection des phrases.

7 Conclusion et perspectives

Dans cet article nous avons proposé une définition générale et opérationnelle de la relation de la comparabilité entre des corpus monolingues annotés. Nous avons entre autres fourni une mesure de la relation de comparabilité ancrée dans un contexte applicatif mais indépendante d'un domaine en particulier, à savoir la comparaison en termes de test statistique de significativité. Nous avons aussi énoncé une procédure de construction d'un corpus comparable.

Nos expérimentations de ces propositions se sont réalisées autour de la construction d'un corpus comparable annoté morpho-syntaxiquement. Nous avons montré notamment que la mesure de la perplexité définie dans la théorie de l'information constitue un moyen de prioriser les phrases à sélectionner pour construire un corpus comparable. Pour une quantité équivalente de phrases d'entraînement, la précision est plus élevée avec un système entraîné sur des phrases sélectionnées selon leur similarité avec un corpus source que pour un système entraîné avec des phrases extraites «aléatoirement». La quantité de phrases requise est une question importante car elle a une incidence sur la taille des modélisations construites, et indirectement sur le choix des modélisations qui peuvent être embarquées dans les systèmes à ressources limitées. Nous montrons néanmoins que la procédure pour exploiter la perplexité a son importance et qu'elle doit tenir compte du corpus source et du corpus relais dans sa globalité. Plus généralement, nous montrons que la quantité de données annotée automatiquement pour entraîner un système n'est pas un critère suffisant. Il apparaît important de le discuter en fonction de la taille du corpus source et du type d'analyse à projeter.

Au sujet du processus de construction du corpus comparable, notre idée fut de favoriser la sélection de phrases d'un corpus relais qu'un modèle de langue, construit sur un corpus source, prédit avec le moins de surprise. Notre approche de la sélection des phrases est sans supervision macroscopique. Les résultats de l'expérience décrite à la section 5.3 tendent à montrer cette faiblesse dans notre approche. En effet, rien ne garantit qu'un tel processus de sélection ne conduise pas à des problèmes de redondance et de représentativité faussée par rapport à la réalité. L'utilisation du critère de pertinence marginale maximale (*Maximal Marginal Relevance*) est une piste possible au filtrage de la redondance tout en maintenant une représentativité des phénomènes (Carbonell & Goldstein, 1998). Plus généralement, nous souhaitons comparer différentes mesures d'ordonnancement des phrases à sélectionner notamment pour chercher à mieux couvrir les propriétés des corpus comparables que nous avons définies.

Enfin, pour poursuivre l'expérience de la section 5.3, nous aimerions étudier la possibilité de construire une mesure de comparabilité fondée sur le rapport de scores de perplexités calculés réciproquement à partir de deux corpus distincts.

Remerciements

Ce travail qui s'inscrit dans le cadre du projet CRISTAL `www.projet-cristal.org` a bénéficié d'une aide de l'Agence National de la Recherche portant la référence ANR-12-CORD-0020. Nous remercions aussi les relecteurs pour leurs commentaires avisés.

Références

ABEILLÉ A. & BARRIER N. (2004). Enriching a french treebank. In Actes de la conférence LREC, Lisbonne.

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building and using Parsed Corpora, chapter Building a treebank for French. Language and Speech series : Kluwer, Dordrecht.

BO L., GAUSSIER E. & AIZAWA A. (2011). Clustering comparable corpora for bilingual lexicon extraction. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, p. 473–478.

BOUDIN F. & HERNANDEZ N. (2012). Détection et correction automatique d'erreurs d'annotation morpho-syntaxique du french treebank. In Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN, p. 281–291, Grenoble, France : ATALA/AFCP.

CARBONELL J. & GOLDSTEIN J. (1998). The use of mmr, diversity-based reranking for reordering documents and producing summaries. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '98, p. 335–336, New York, NY, USA : ACM.

CRABBÉ B. & CANDITO M. (2008). Expériences d'analyse syntaxique statistique du français. In Actes de la 15ème conférence sur le Traitement Automatique des Langues Naturelles (TALN), Avignon, France.

DÉJEAN H. & ÉRIC GAUSSIER (2002). Une nouvelle approche à l'extraction de lexiques bilingues à partir de corpus comparables. Lexicometrica, Numéro thématique "Alignement lexical dans les corpus multilingues", p. 1–22.

HERNANDEZ N. & BOUDIN F. (2013). Construction d'un large corpus écrit libre annoté morpho-syntaxiquement en français. In Actes de la 20e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2013), p. 160–173, Les Sables d'Olonne, France.

KNESER R. & NEY H. (1995). Improved backing-off for m-gram language modeling. In International Conference on Acoustics, Speech, and Signal Processing.

KOEHN P. (2005). Europarl : A parallel corpus for statistical machine translation. In MT Summit.

MCENERY A. M. & XIAO R. Z. (2007). Parallel and comparable corpora : What are they up to ?, In Incorporating Corpora : Translation and the Linguist. Translating Europe. Multilingual Matters.

MEDLOCK B. (2006). An introduction to nlp-based textual anonymisation. In Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC-2006), Genoa, Italy : European Language Resources Association (ELRA). ACL Anthology Identifier : L06-1110.

NIELSEN M. (2011). Reinventing Discovery : The New Era of Networked Science. Princeton University Press.

PAULS A. & KLEIN D. (2011). Faster and smaller n-gram language models. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies - Volume 1, HLT '11, p. 258–267, Stroudsburg, PA, USA : Association for Computational Linguistics.

REHBEIN I. (2011). Data point selection for self-training. In Proceedings of the Second Workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL 2011), Dublin, Ireland.

SØGAARD A. (2011). Data point selection for cross-language adaptation of dependency parsers. In The 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies, Portland, Oregon.

TOUTANOVA K., KLEIN D., MANNING C. & SINGER Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In Proceedings of the 3rd Conference of the North American Chapter of the ACL (NAACL 2003), p. 173–180 : Association for Computational Linguistics.

WANG R. & CALLISON-BURCH C. (2011). Paraphrase fragment extraction from monolingual comparable corpora. In Proceedings of the 4th Workshop on Building and Using Comparable Corpora : Comparable Corpora and the Web, p. 52–60, Portland, Oregon : Association for Computational Linguistics.

Vers une approche simplifiée pour introduire le caractère incrémental dans les systèmes de dialogue

Hatim Khouzaimi[1,2] Romain Laroche[1] Fabrice Lefèvre[2]
(1) Orange Labs, 38-40 rue du Général Leclerc 92794 Issy-les-Moulineaux, France
(2) Laboratoire Informatique d'Avignon, 339 chemin des Meinajaries 84911 Avignon, France
hatim.khouzaimi@orange.com, romain.laroche@orange.com, fabrice.lefevre@univ-avignon.fr

Résumé. Le dialogue incrémental est au cœur de la recherche actuelle dans le domaine des systèmes de dialogue. Plusieurs architectures et modèles ont été publiés comme (Allen *et al.*, 2001; Schlangen & Skantze, 2011). Ces approches ont permis de comprendre différentes facettes du dialogue incrémental, cependant, les implémenter nécessite de repartir de zéro car elles sont fondamentalement différentes des architectures qui existent dans les systèmes de dialogue actuels. Notre approche se démarque par sa réutilisation de l'existant pour tendre vers une nouvelle génération de systèmes de dialogue qui ont un comportement incrémental mais dont le fonctionnement interne est basé sur les principes du dialogue traditionnel. Ce papier propose d'intercaler un module, appelé *Scheduler*, entre le service et le client. Ce *Scheduler* se charge de la gestion des événements asynchrones, de manière à reproduire le comportement des systèmes incrémentaux vu du client. Le service, de son côté, ne se comporte pas de manière incrémentale.

Abstract. Incremental dialogue is at the heart of current research in the field of dialogue systems. Several architectures and models have been published such as (Allen *et al.*, 2001; Schlangen & Skantze, 2011). This work has made it possible to understand many aspects of incremental dialogue, however, in order to implement these solutions, one needs to start from scratch as the existing architectures are inherently different. Our approach is different as it tends towards a new generation of incremental systems that behave incrementally but work internally in a traditional way. This paper suggests inserting a new module, called the *Scheduler*, between the service and the client. This *Scheduler* manages the asynchronous events, hence reproducing the behaviour of incremental systems from the client's point of view. On the other end, the service does not work incrementally.

Mots-clés : Systèmes de Dialogue, Traitement Incrémental, Architecture des Systèmes de Dialogue.

Keywords: Dialogue Systems, Incremental Processing, Dialogue Systems Architecture.

1 Introduction

Les systèmes de dialogue traditionnels [1] fonctionnent au tour par tour. L'utilisateur parle et quand il se tait, il donne la parole au système. Certains systèmes permettent à l'utilisateur de les interrompre pour éviter les désynchronisations (*barge-in*) mais sans relier le moment et le contenu de son intervention avec la phrase du système et sans pouvoir continuer à parler en ignorant certaines interruptions (confirmations, bruit...). Ce modèle de dialogue a certes l'avantage d'être simple, mais il est loin de la réalité du dialogue naturel entre humains (Edlund *et al.*, 2008). Quand ceux-ci interagissent entre eux, ils se comprennent au fur et à mesure qu'ils parlent, peuvent s'interrompre mutuellement et peuvent même deviner la fin d'une proposition avant que celle-ci ne soit totalement prononcée par la personne qui parle (Tanenhaus *et al.*, 1995; Brown-Schmidt & Hanna, 2011; DeVault *et al.*, 2011). Des travaux ont également mis en valeur un procédé par étapes, lors de la construction du sens pendant la lecture (Ilkin & Sturt, 2011) et d'autres documents plus généraux en psycholinguistique évoquent ce phénomène (Levelt, 1989; Clark, 1996).

Intégrer ce genre de comportement dans les systèmes de dialogue permet d'avoir des systèmes plus réactifs et qui offrent une expérience utilisateur potentiellement plus agréable car plus proche du dialogue homme-homme. On parle de dialogue incrémental. De nombreuses études ont montré la supériorité des stratégies de dialogue incrémentales en termes de

1. Tout au long de ce papier, nous utiliserons l'adjectif *traditionnel* pour qualifier les systèmes de dialogue non-incrémentaux.

satisfaction utilisateur (Skantze & Schlangen, 2009; Baumann & Schlangen, 2013; El Asri *et al.*, 2014) et de complétion de tâche (Matthias, 2008; El Asri *et al.*, 2014).

Le terme *incrémental* a été utilisé initialement dans le domaine de l'informatique. Un compilateur incrémental (Lock, 1965) compile chaque ligne indépendamment des autres, ainsi, une modification locale ne peut pas affecter la compilation globale. (Wirén, 1992) utilise pour la première fois cette notion pour l'analyse du langage naturel. Les données en entrée d'un module incrémental ne lui sont pas communiquées en un seul bloc mais elles sont divisées en plusieurs morceaux. Dès le premier fragment, ce module commence déjà son traitement et cela donne lieu à des sorties hypothétiques disponibles dès qu'elles sont calculées.

Un système de dialogue incrémental fonctionne selon le même principe. La requête de l'utilisateur est divisée en *unités incrémentales* (Schlangen & Skantze, 2011) (e. g. division temporelle du signal audio pour les systèmes vocaux, division en mots pour les systèmes texte...) qui sont envoyés à la suite au système. Ce dernier maintient une hypothèse de réponse qui évolue au fur et à mesure avec l'arrivée de nouvelles informations. Cette évolution peut être visible par l'utilisateur ou cachée. Dans le cas des systèmes multimodaux, les canaux autres que la parole peuvent être exploités pour faire un *feedback* à l'utilisateur pendant qu'il parle (Fink *et al.*, 1998) (par exemple, un avatar qui hoche la tête quand il comprend une nouvelle information et fronce les sourcils quand il détecte une incohérence...). Par ailleurs, un système de dialogue incrémental reste à l'écoute de l'utilisateur même quand il prend la parole. Ainsi, outre la réactivité, un des avantages majeurs des stratégies de dialogue incrémentales réside dans la possibilité pour l'utilisateur d'interrompre le système (Matsuyama *et al.*, 2009; Selfridge *et al.*, 2013) quand il pense que celui-ci a mal compris sa requête. Contrairement au *barge-in* autorisé par certains systèmes traditionnels, le contenu et l'instant de l'intervention sont mis en relation avec l'énoncé interrompu pour en dégager le sens. De plus, le système peut choisir de ne pas réagir à certaines interruptions (comme les confirmations par exemple). Par conséquent, l'intervention de l'utilisateur peut ne porter que sur un fragment de la réponse du système comme c'est le cas dans les stratégies d'énumération (El Asri *et al.*, 2014). Il peut donc rattraper une désynchronisation plus rapidement et plus facilement. En plus, suivant le sens donné à l'interruption, celle-ci peut donner lieu à des actions différentes ou peut être ignorée.

Plusieurs travaux proposant des architectures de systèmes de dialogue incrémentaux existent déjà. Néanmoins, ces architectures présupposent de construire de tels systèmes en partant de rien, ce qui constitue un investissement considérable. Un système de dialogue est composé de plusieurs modules constituant *la chaîne de dialogue*, qui généralement se présente comme suit : la reconnaissance vocale, le traitement du langage naturel, la gestion du dialogue, la génération de langage naturel puis la synthèse de la parole. Pour rendre un tel système incrémental, des travaux antérieurs proposent de rendre la plupart de ses modules incrémentaux. Par opposition, notre démarche vise à transformer un système de dialogue traditionnel en système incrémental à moindre coût, en procédant par étapes.

La section 2 décrit l'état de l'art concernant les systèmes de dialogue incrémentaux. Dans le cadre de ce papier, la solution proposée consiste à introduire un module entre le client et le service de dialogue : le *Scheduler*. Ce concept sera introduit en Section 3. En Section 4, la méthode sera appliquée à un assistant texte de recherche de contenu puis à un système vocal de dictée de numéros : DictaNum. Enfin, une discussion est menée en Section 5 et la Section 6 conclut ce papier par des perspectives d'amélioration.

2 Travaux précédents

Nous distinguons quatre catégories de systèmes de dialogue suivant leur niveau d'intégration du caractère incrémental. La première concerne les systèmes traditionnels (Laroche *et al.*, 2011). La seconde englobe les systèmes de dialogue traditionnels avec quelques stratégies isolées (El Asri *et al.*, 2014) et la troisième se compose des systèmes qui reproduisent un comportement incrémental tout en gardant un fonctionnement interne traditionnel (Selfridge *et al.*, 2012; Hastie *et al.*, 2013). Enfin, les systèmes incrémentaux dont les composants internes le sont également constituent le dernier groupe (Dohsaka & Shimazu, 1997; Allen *et al.*, 2001; Schlangen & Skantze, 2011). La figure 5, discutée en section 5, propose une comparaison entre ces catégories en termes de fonctionnalités offertes par chacune.

NASTIA (El Asri *et al.*, 2014) est un système de dialogue destiné à la prise de rendez-vous avec un technicien pour une intervention à domicile. Ce système vient à la suite des travaux menés lors du projet européen CLASSiC (Laroche & Putois, 2010). Il interagit avec l'utilisateur en utilisant plusieurs stratégies de dialogue destinées à la récupération d'un créneau durant lequel le technicien peut se déplacer. Parmi celles-ci, la stratégie nommée *List of Availabilities* (LA) consiste à lister les créneaux disponibles et à attendre que l'utilisateur interrompe le système quand il entend une option qui l'intéresse. Cette dernière stratégie relève du dialogue incrémental car elle ne fonctionne pas en tour de parole.

L'expérimentation menée dans (El Asri *et al.*, 2014) montre qu'elle permet d'avoir un gain de près de 10% de taux de complétion de tâche ainsi qu'une amélioration significative de l'expérience utilisateur.

PARLANCE (Hastie *et al.*, 2013) est un exemple de système de dialogue appartenant à la troisième catégorie. Il a été développé au sein du projet européen du même nom. Son architecture conserve les même modules qu'une architecture classique mais avec quelques fonctionnalités en plus pour supporter un comportement incrémental : "the PARLANCE system architecture [...] maintains the modularity of a traditional SDS while at the same time allowing for complex interaction at the micro-turn level between components". Le principal module qui fait la différence avec les autres architectures est le MIM (Micro-turn Interaction Manager). Il gère les prises de parole du système, les périodes d'écoute et la génération des backchannels, tout ceci à l'échelle du micro-tour. L'architecture la plus proche de celle proposée ici est introduite dans (Selfridge *et al.*, 2012). Un module est introduit entre la reconnaissance vocale incrémentale et la synthèse de parole d'un côté et un service de dialogue de l'autre. Néanmoins, il ne s'agit pas de l'idée centrale du papier qui présente cette démarche comme étant un travail préliminaire pour pouvoir simuler un dialogue incrémental à moindre coût, et ainsi faire une expérimentation illustrant d'autres points qui ne nous intéressent pas ici. Dans cet article, nous avons choisi d'étudier cette démarche dans le détail, d'y apporter une formalisation, de l'ancrer dans la littérature existante et de prouver son fonctionnement à travers deux implémentations.

L'architecture proposée dans (Dohsaka & Shimazu, 1997) contient huit modules fonctionnant en parallèle : le Speech Recognizer, le Response Analyzer, le Dialogue Controller, le Problem Solver, l'Utterance Planner, l'Utterance Controller, le Speech Synthesizer et le Pause Monitor. La requête de l'utilisateur est vue comme un problème que l'utilisateur soumet au système. Celle-ci est captée par le Speech Recognizer, transmise sous forme de texte au Response Analyzer qui en dégage des concepts compréhensibles par le Dialogue Controller. Ce-dernier communique à la fois avec le Problem Solver et l'Utterance Planner qui délivre une réponse à l'utilisateur à travers l'Utterance Controller et le Speech Synthesizer. L'objectif est de pouvoir commencer à proposer une solution au problème alors qu'il est toujours en cours de résolution. Cette architecture appartient à la dernière catégorie de systèmes de dialogue sur l'échelle relative au caractère incrémental. De même, dans (Allen *et al.*, 2001), l'architecture proposée se compose de trois modules principaux : l'Interpretation Manager, le Behavioral Agent et le Generation Manager. Chaque nouvelle requête de l'utilisateur est captée par l'Interpretation Manager. Celui-ci diffuse cette information dans le système de manière incrémentale. Le Behavioral Agent est chargé de gérer le plan d'action du système et le Generation Manager s'occupe des interactions avec l'utilisateur. Tous deux sont construits de façon à agir de façon incrémentale compte-tenu des entrées venant de l'Interpretation Manager.

Une architecture plus générale est décrite dans (Schlangen & Skantze, 2011). Un système de dialogue peut être vu dans le cadre général comme une chaîne de modules séparés ayant chacun une tâche précise. Pour réaliser un système incrémental, (Schlangen & Skantze, 2011) part de ce postulat de départ et sépare chacun de ces modules en 3 parties : le Left Buffer, l'Internal State et le Right Buffer. Le Left Buffer représente l'entrée du module, l'Internal State désigne l'état de celui-ci et le Right Buffer contient la sortie. L'information est transportée et propagée dans le système sous forme d'IUs (Incremental Units). Par exemple, toutes les 500 ms, une nouvelle IU sous forme de signal sonore est placée dans le Left Buffer de la reconnaissance vocale (ASR) qui la transforme en IU au format texte. Le Right Buffer d'un module est le Left Buffer du suivant ce qui assure la propagation de l'information. Ce modèle générique englobe les systèmes de dialogue des quatre catégories, les systèmes non-incrémentaux étant perçus comme des cas particuliers de systèmes incrémentaux : "we can now see that a non-incremental system can be characterised as a special case of an incremental system, namely one where IUs are always maximally complete [...] and where all modules update in one go".

L'architecture proposée ici fait partie de la troisième catégorie de notre classification. Par comparaison à PARLANCE, le *Scheduler* a un rôle similaire au MIM, cependant, il est construit pour s'interfacer avec une architecture traditionnelle préexistante, rajoutant ainsi une couche au service pour simuler un comportement incrémental. Le MIM, quant à lui, s'insère dans une architecture qui est faite de façon à communiquer avec lui (plusieurs autres modules s'interfacent directement avec lui), jouant ainsi un rôle central dans le système de dialogue. L'avantage de ce type de systèmes réside dans le fait qu'ils se comportent de manière incrémentale tout au long du dialogue (contrairement aux systèmes des deux premières catégories) et sont donc plus réactifs, plus naturels tout en présentant moins de problèmes de désynchronisation. En outre, par opposition à la dernière catégorie, la conception de tels systèmes est moins coûteuse car elle permet de partir de systèmes non-incrémentaux existants et de les rendre incrémentaux. Dans ce qui suit, nous formalisons notre approche en utilisant les concepts introduits dans le modèle général et abstrait de (Schlangen & Skantze, 2011).

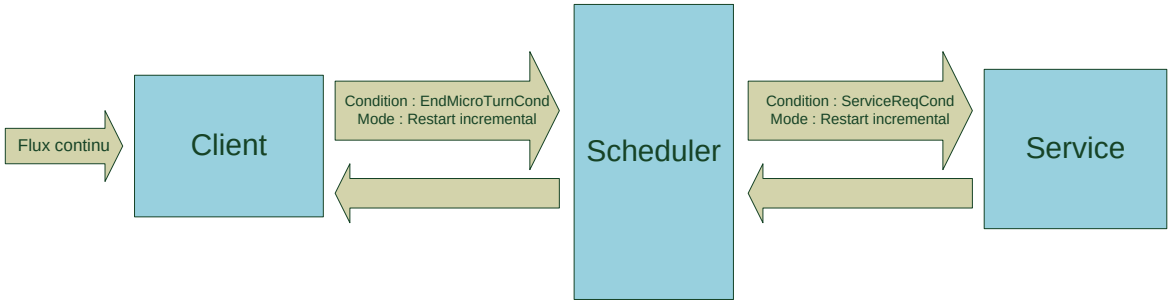


FIGURE 1 – Le *Scheduler* : module intermédiaire entre le client et le système de dialogue.

3 Le Scheduler

En règle générale, les systèmes de dialogue traditionnels se composent d'un client installé sur le terminal de l'utilisateur et d'un service déployé sur un serveur. Ils adoptent une approche tour par tour. Nous appellerons *tour de dialogue* l'intervalle de temps durant lequel l'utilisateur prend la parole une fois puis le système lui répond. Ainsi, le temps du dialogue peut être vu comme un enchaînement de tours de dialogue : T^1, T^2,..., T^k... Au tour T^k, le client envoie une requête complète Req^k au service qui la traite et renvoie la réponse Rep^k correspondante (il peut éventuellement effectuer un autre traitement en plus comme la modification d'une base de données). Par conséquent, ce tour peut lui-même être divisé en deux intervalles temporels, le *tour utilisateur* et le *tour système* $T^k = T^{k,U} \cup T^{k,S}$. En partant d'un tel système, et sans aucune autre hypothèse supplémentaire, nous allons montrer comment le rendre incrémental. La méthode consiste à intercaler un module intermédiaire entre le client et le service : le *Scheduler* (cf. Figure 1). Le rôle de celui-ci sera de simuler un comportement incrémental vu du client (le terme *Scheduler* est emprunté à (Laroche, 2010)). Cette architecture est une instanciation du modèle abstrait proposé dans (Schlangen & Skantze, 2011). Le client, le *Scheduler* et le service en constituent les modules. Les deux premiers sont incrémentaux alors que le troisième ne l'est pas. Il n'est pas pertinent pour nous de parler de Left Buffer et de Right Buffer, les liens entre les modules sont vus comme des canaux d'information (réseau dans notre implémentation). Dans la suite de cette section, nous décrirons les comportements traditionnels du client et du service, avant d'aborder les modifications qui leurs sont apportées dans notre nouvelle architecture.

3.1 Rôle du client et du service dans une architecture traditionnelle

Le client reçoit un signal provenant de l'utilisateur. Généralement, celui-ci se présente sous forme d'un flux continu d'information (signal audio en vocal, suite de caractères en texte...). Si cela n'est pas vérifié (par exemple, une interface web où chaque bouton désigne une requête), la notion de dialogue incrémental n'a pas de sens. Ainsi, le client doit avoir une condition $EndTurnCond$ (End of Turn Condition) lui permettant de savoir à quel moment envoyer la requête au service. Cette condition correspond généralement à un silence suffisamment long (Raux & Eskenazi, 2008; Wlodarczak & Wagner, 2013) pour les systèmes vocaux et à un retour chariot pour les systèmes textuels. Nous appellerons *instant d'activation* d'une condition le moment où elle passe de $faux$ à $vrai$. Un dialogue traditionnel est une alternance de tours utilisateur dont la fin est déterminée par l'instant d'activation de $EndTurnCond$ et de tours du système qui se terminent quand celui-ci redonne la parole à l'utilisateur. Il se peut que ce dernier ne dise rien durant le temps qui lui est imparti, nous appellerons tout de même cela un tour de parole.

Le service se compose d'une interface avec le client, d'un contexte interne et d'une interface avec le monde extérieur. L'interface avec le client gère la communication avec celui-ci ainsi que la compréhension de ses requêtes et le contexte interne correspond à toutes les informations dont dispose le service sur l'état du dialogue à chaque instant. L'interface avec le monde extérieur lui permet d'envoyer des requêtes et d'agir sur des modules externes au système de dialogue (bases de données ou appareils en domotique par exemple).

3.2 Passage en mode incrémental

Pour rendre un tel système incrémental, nous modifions la façon dont les requêtes sont envoyées. Une nouvelle condition définit l'envoi d'une nouvelle requête de la part du client. Elle est notée $EndMicroTurnCond$ (End of Micro-Turn

Condition, cf. Figure 1) et est moins restrictive que $EndTurnCond$: $EndTurnCond$ implique $EndMicroTurnCond$ (le client envoie des requêtes plus souvent). Nous appellerons *micro-tour utilisateur* l'intervalle temporel compris entre deux instants d'activation de $EndMicroTurnCond$. $T^{k,U}$ peut être divisé en $n^{k,U}$ micro-tours utilisateur $\mu T_i^{k,U}$: $T^{k,U} = \bigcup_{i=1}^{n^{k,U}} \mu T_i^{k,U}$. Nous définissons également le *sous-tour utilisateur* $T_p^{k,U} = \bigcup_{i=1}^{p} \mu T_i^{k,U}$ où $1 \leq p \leq n^{k,U}$. Nous manipulons des intervalles temporels, d'où le choix de l'opérateur d'union. En général, $EndMicroTurnCond$ correspond à un cycle d'horloge précis (nouveau micro-tour toutes les 500 ms...) ou à l'arrivée d'une nouvelle information (modification de la sortie de l'ASR par exemple). Notons également qu'à chaque instant d'activation de $EndTurnCond$, le client envoie un signal dédié noté *signal_ETC* pour signaler cet événement au *Scheduler*.

Le *Scheduler* est un module qu'on propose de placer entre le client et le service. Cet intermédiaire a pour rôle de rendre l'ensemble {Scheduler + Service} équivalent à un système incrémental vu du client sans modifier le fonctionnement du service. À chaque micro-tour utilisateur, il reçoit une nouvelle entrée venant du client. Le *Scheduler* est également muni d'une condition $ServiceReqCond$ (Service Request Condition, cf. Figure 1) sous laquelle il transmet la requête venant du client au service, récupère la réponse correspondante et la stocke pour qu'elle puisse être récupérée ultérieurement par le client. Cette condition peut être vraie tout le temps, auquel cas, à chaque requête provenant du client, le service est sollicité pour obtenir une réponse. Une façon simple et plus économe en nombre de requêtes est de définir $ServiceReqCond$ comme l'arrivée d'une requête différente de la précédente. Si le *Scheduler* intervient juste après l'ASR, il pourra vérifier que le texte correspondant à la requête a changé (Si $EndMicroTurnCond$ inclut déjà cette condition, il est inutile de la rajouter à $ServiceReqCond$). De même, s'il intervient après l'analyse sémantique, la vérification se portera sur l'arrivée d'un nouveau concept.

L'autre rôle clé du *Scheduler* est de décider du moment où il engage le système à valider l'hypothèse de requête en cours et à s'en tenir à elle (en n'attendant plus de nouvelles informations de la part du client pour venir la confirmer ou l'infirmer). Ceci marque la fin du tour en cours et correspond à la notion de *commit* (Schlangen & Skantze, 2011) décrite en détail dans la section 3.3. On notera $CommitCond$ (Commit Condition) la condition sous laquelle le *Scheduler* prend cette décision. Par exemple, si l'utilisateur doit fournir son numéro de téléphone à 10 chiffres au service, on pourra prendre $CommitCond = (length(num) == 10)$ où $length(num)$ est la longueur du numéro reconnu à chaque instant. Il est important de noter que dans le cadre incrémental proposé, c'est l'instant d'activation de $CommitCond$ qui marque la fin d'un tour et non pas celui de $EndTurnCond$ (le *Scheduler* peut décider d'effectuer un *commit* sans recevoir de *signal_ETC* de la part du client). Néanmoins, $EndTurnCond$ implique $CommitCond$.

À chaque nouveau tour utilisateur $T^{k,U}$, l'utilisateur formule une nouvelle requête et au micro-tour $\mu T_i^{k,U}$, celle-ci n'est pas encore accessible dans son intégralité (sauf si $i = n^{k,U}$), néanmoins, le système dispose déjà d'une hypothèse provisoire qui sera appelée *sous-requête* et notée Req_i^k. Celle-ci sera envoyée au *Scheduler* à chaque micro-tour utilisateur. Le fait d'envoyer toute la requête depuis le début du tour utilisateur en cours correspond au mode *restart incremental* introduit dans (Schlangen & Skantze, 2011). Remarquons au passage que si $i_1 < i_2$ alors $Req_{i_1}^k$ n'est pas obligatoirement un préfixe de $Req_{i_2}^k$ (en vocal, l'arrivée d'un nouvel incrément de signal audio au niveau de l'ASR peut changer l'hypothèse de sortie et pas seulement la compléter).

Le client est composé de deux processus indépendants : le premier se charge de l'envoi des requêtes vers le *Scheduler* en suivant le fonctionnement décrit plus haut, et le second récupère les réponses stockées dans ce-dernier. La récupération de ces réponses se fait à une fréquence de l'ordre de la fréquence des micro-tours utilisateur afin de s'assurer que le client est en permanence à jour (qu'il a récupéré la dernière réponse du service disponible dans le *Scheduler*). Dans le cas des systèmes vocaux, le *Scheduler* a pour tâche de déterminer quelles réponses du système devraient être prononcées par la synthèse vocal et lesquelles devraient être ignorées. Pour cela, il ajoute un marqueur dédié aux réponses qu'il sélectionne avant qu'elles ne soient récupérées par le client. Cela permet au *Scheduler* de gérer les sorties système de manière incrémentale.

Tout l'intérêt de la solution proposée ici est que le service reste quasiment inchangé au niveau fonctionnel (quelques modifications au niveau applicatif peuvent s'avérer nécessaires, voir la section 4.2 pour un exemple). Le seul changement concerne le maintien d'un double contexte : le contexte réel et le contexte simulé (voir la section 3.3). À chaque fois que $ServiceReqCond$ est vraie, le *Scheduler* envoie au service une requête incomplète et récupère la réponse. Ainsi, il permet au client (si celui-ci choisit de récupérer les réponses intermédiaires) de voir ce qu'aurait répondu le service avant que la requête ne soit totalement formulée. Le fait de conserver un comportement non-incrémental du service et d'avoir besoin de toutes les IU depuis le début de la requête à chaque fois que ce-dernier est sollicité justifie le choix du mode *restart incremental*.

Au même titre que le *Scheduler*, le service peut également ordonner à ce dernier d'effectuer un *commit*. Une telle action doit être remontée au *Scheduler* pour l'informer qu'il ne peut pas remplacer la dernière hypothèse de requête par une autre (voir la notion de *rollback* décrite dans la section 3.3). La relation de *grounding* présentée dans (Schlangen & Skantze, 2011) permet d'avoir un tel mécanisme. À la sortie d'un module, chaque IU connaît l'ensemble des IU en entrée qui l'ont engendrée : on dit qu'elle est basée sur ces IU (*grounded in*). Ainsi, quand le module décide de valider une IU par un *commit*, cette validation se propage à toutes les IU sur laquelle elle est basée et ainsi de suite.

Enfin, tout comme nous avons défini la notions de micro-tour utilisateur, nous faisons de même du côté du système. Dans un cadre classique, la synthèse vocale (TTS) joue la réponse du système durant chaque tour système $T^{k,S}$. En dialogue incrémental, cette durée peut être divisée en $n^{k,S}$ micro-tours système $\mu T_i^{k,S}$: $T^{k,S} = \bigcup_{i=1}^{n^{k,S}} \mu T_i^{k,S}$. Cette division est déterminée par le service au moment où il envoie sa réponse (cf. 4.2 pour un exemple). Quand l'utilisateur interrompt le système, le timing de son interruption est donné par le micro-tour système durant lequel il est intervenu. Par ailleurs, au moment du *barge-in*, on passe au tour suivant T^{k+1}. Remarquons que cela ne concerne que les systèmes vocaux, les systèmes textuels ne pouvant être interrompus.

3.3 Commit, rollback et double contexte

Dans le cadre incrémental, le système formule des hypothèses de réponse au fur et à mesure que l'utilisateur lui fournit des informations en entrée en complétant sa requête. Ces hypothèses fluctuent en fonction des nouvelles entrées, de la base des connaissances actuelles ou d'autres paramètres du système. Cependant, au bout d'un moment, le système doit effectuer une action qui engage le dialogue dans la direction imposée par la dernière hypothèse faite. Par exemple, si le système commence à formuler une réponse ou à modifier une base de données, il ne peut pas se permettre d'ignorer son hypothèse et de la modifier car celle-ci a donné naissance à une action concrète perceptible par l'utilisateur. À partir de ce moment là, on dit que le système a effectué un *commit* de sa dernière hypothèse.

L'opération inverse du *commit* sera appelée *rollback* (terme emprunté au lexique de la gestion de base de données). Tant que le *commit* d'une hypothèse n'est pas effectif, le système peut décider d'oublier cette hypothèse et de revenir à l'état dans lequel il était au moment du dernier *commit*.

Certaines requêtes du *Scheduler* peuvent être amenées à modifier le contexte du dialogue or cela n'est pas l'effet recherché. Ce module est censé communiquer avec le système de dialogue principalement pour voir ce qu'il aurait répondu à la requête à un instant donné (la plupart du temps encore incomplète). Pour remédier à cela, une solution consiste à maintenir deux contextes au sein des systèmes de dialogue : un contexte *réel* qui est l'équivalent du contexte traditionnel, sauvegardé à chaque *commit*, et un contexte *de simulation* avec lequel travaillera le *Scheduler* pour obtenir les résultats intermédiaires du système.

Cette notion de double contexte est indissociable des idées de *commit* et *rollback*. Quand le *Scheduler* décide d'effectuer un *commit*, **il copie le contexte de simulation dans le contexte réel**. À l'inverse, il fait le *rollback* d'une hypothèse quand **il copie le contexte réel dans le contexte de simulation**. Il oublie ainsi ce qui s'est passé depuis le dernier *commit*.

À chaque micro-tour utilisateur, le client envoie au *Scheduler* tout ce que l'utilisateur a prononcé (ou tapé) depuis le dernier *commit*. Le *Scheduler* envoie ensuite cette requête (encore incomplète) au service et récupère la réponse. Si au prochain micro-tour, le *Scheduler* ne décide pas d'effectuer un *commit* mais souhaite plutôt envoyer une nouvelle requête, alors un ordre de *rollback* est envoyé au service avant cette requête, comme l'envoi se fait en mode *restart incremental* (dans le cadre de ce papier, l'opération de *rollback* intervient uniquement dans ce cas là). Les Figures 2 (diagramme de séquence) et 3 illustrent ce fonctionnement. Sur la Figure 2, les conditions $EndTurnCond$, $EndMicroTurnCond$, $ServiceReqCond$ et $CommitCond$ sont écrites à gauche des flux qu'ils génèrent. Les zones d'activité et d'inactivité du client concernent le processus d'envoi alors que les flèches en pointillées désignent les retours vers le processus de récupération des réponses. Ceux-ci ne sont pas synchronisés avec le reste des flux, même si, par souci de clarté, ils le sont sur le diagramme. Par ailleurs, sur cette figure, la décision de *commit* a été prise suite à un *signal_ETC* or ce n'est pas toujours le cas.

Nous notons $ctxt(T^k)$ le contexte réel obtenu à l'issue du tour T^k, $ctxt(T^0)$ étant le contexte initial au début du dialogue. Ce contexte fluctue durant les tours utilisateur mais reste fixe durant les tours système ($ctxt(T^k) = ctxt(T^{k,U})$). Au moment du *commit* qui marque la fin du tour T^k, le contexte réel prend la valeur du contexte simulé à cet instant là : $ctxt(T^k) = ctxt(T^{k-1} + T^{k,U}_{n^{k,U}})$.

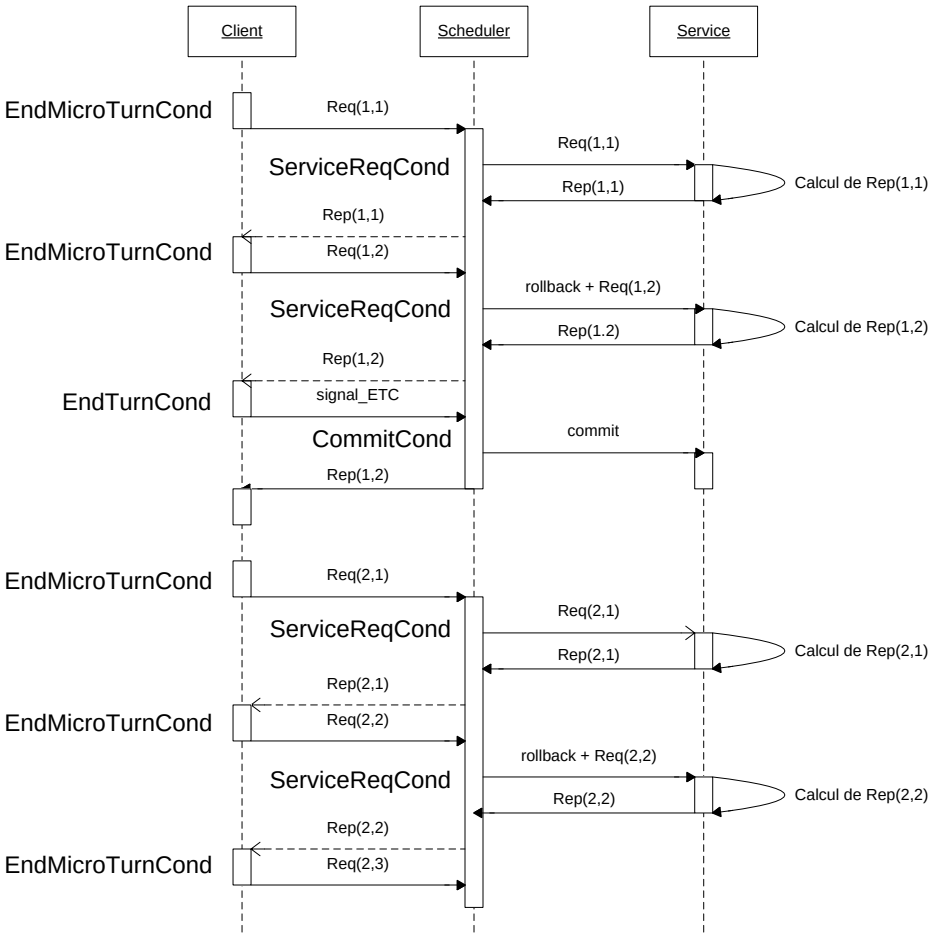


FIGURE 2 – Principe de fonctionnement du *Scheduler* (les flèches en pointillées désignent les flux vers le processus de récupération dans le client).

4 Implémentations

Dans le but de montrer la faisabilité de cette solution, deux systèmes de dialogue existants (développés à Orange Labs) ont été rendus incrémentaux à l'aide de l'introduction d'un *Scheduler*. Le premier est un service texte où le client se présente sous forme d'interface web et le second est un service vocal où l'utilisateur interagit avec une application Android. Avec un minimum de modifications, un comportement incrémental a été intégré à ces deux exemples ce qui montre la facilité d'implémentation et d'adaptation de la solution tout en démontrant le comportement incrémental du résultat obtenu, à la fois dans le cadre des stratégies implémentées et des modalités d'interaction utilisées (mode texte et interface web d'un côté et mode vocal de l'autre).

4.1 CFAsT : Content Finder AssistanT

Le CFAsT est une application permettant de générer automatiquement et rapidement un assistant de navigation en mode texte à partir d'une base de contenus donnée. Il a été développé au sein d'Orange Labs et les services qu'il génère se composent d'un service web déployé sur un serveur et d'une interface web où l'utilisateur peut entrer du texte (en langage naturel) ou appuyer sur des boutons (propositions de mots-clés ou de cibles).

Tour	Sous-tour utilisateur	Entrée	Contexte réel	Contexte simulé
T^1	$T_1^{1,U}$	Req_1^1	ctxt(T^0)	ctxt($T^0 + T_1^{1,U}$)
	$T_2^{1,U}$	Req_2^1	ctxt(T^0)	ctxt($T^0 + T_2^{1,U}$)
	...	...	ctxt(T^0)	...
	$T_{n^{1,U}}^{1,U}$	$Req_{n^{1,U}}^1$	ctxt(T^0)	ctxt($T^0 + T_{n^{1,U}}^{1,U}$)
	COMMIT : $ctxt(T^1) = ctxt(T^0 + T_{n^{1,U}}^{1,U})$			
T^2	$T_1^{2,U}$	Req_1^2	ctxt(T^1)	ctxt($T^1 + T_1^{2,U}$)
	...	...	ctxt(T^1)	...

FIGURE 3 – Un double contexte : le contexte réel et le contexte simulé.

Initialement, l'utilisateur est invité à entrer sa requête en langage naturel ou à choisir parmi un certain nombre de mots-clés. Ensuite, il se voit proposer des contenus en plus des mots-clés. Dans le cadre de cette étude, la base de contenus choisie est la FAQ de l'ANPE (Agence Nationale Pour l'Emploi). Un contenu correspond à un ensemble question/réponse de la FAQ. L'utilisateur peut rentrer sa requête en langage naturel ou sélectionner un des trois mots-clés qui lui sont présentés, ensuite, en plus de l'entrée texte, cinq propositions de contenus sont faites ainsi que trois nouvelles propositions contextuelles de mots-clés. L'interaction s'arrête quand un contenu est sélectionné par l'utilisateur.

Le système maintient une liste de mots-clés qu'il met à jour au fur et à mesure de l'interaction. Ainsi, la requête de l'utilisateur se construit et s'enrichit au fur et à mesure jusqu'à ce que l'utilisateur choisisse un contenu.

Le client de démonstration [2] se compose de deux vues : la première contient l'interface classique et la seconde est mise à jour à chaque fois que l'utilisateur tape un espace (à chaque nouveau mot). Elle représente ce qu'aurait répondu le système si l'utilisateur avait validé la requête à ce stade là. À chaque fois que l'utilisateur valide une requête par un retour chariot (cette validation entraîne un *commit* de la part du *Scheduler*), les deux vues sont identiques. Cette application de notre méthode n'a pas d'utilité pratique mais elle sert à en prouver le fonctionnement. En reprenant les concepts introduits ici : {**EndMicroTurnCond** : appui sur la barre d'espace, **EndTurnCond** : retour chariot, **ServiceReqCond** : EndMicroTurnCond, **CommitCond** : EndTurnCond}.

4.2 DictaNum

DictaNum [3] est un système de dialogue (inspiré de NUMBERS (Skantze & Schlangen, 2009)) qui recueille des numéros et confirme leur bonne compréhension. L'utilisateur interagit avec le service en utilisant un client web. La reconnaissance vocale et la synthèse de la parole sont assurées par l'API Web Speech sur Google Chrome. D'autres systèmes existants permettent de dicter des numéros (téléphone, carte bancaire...) comme *How may I help you ?* (Langkilde *et al.*, 1999).

L'intérêt principal de la version incrémentale est de pouvoir dicter son numéro par morceaux. Par conséquent, le système a été modifié au niveau applicatif. Au départ, le numéro est une chaîne vide et à chaque fois qu'un silence est détecté ($EndTurnCond$), la sortie de l'ASR est concaténée à celle-ci et le système émet un *feedback* en la répétant. L'utilisateur peut ensuite continuer sa dictée ou indiquer qu'il n'est pas d'accord avec ce *feedback* et le corriger en commençant sa phrase par 'Non' (voir l'exemple d'application plus bas). Le service comprendra qu'il faut remplacer le dernier fragment dicté et fera un second *feedback* en commençant par s'excuser : 'Désolé,...'. Cependant, effectuer une dictée par morceaux dans un cadre non-incrémental est un processus qui n'est pas naturel et qui manque de réactivité car la durée des silences est trop importante. L'utilisateur préférera effectuer sa dictée en une seule fois car cela prend moins de temps. Le client a été modifié de façon à ne plus attendre un silence de la part de l'utilisateur pour envoyer la requête. Toutes les 500 ms, il envoie tout ce que l'utilisateur a dit depuis le dernier *commit*. On note Δ_s le seuil de silence dans le cadre non-incrémental et δ_s un nouveau seuil utilisé pour morceler la dictée du numéro ($\delta_s \leq \Delta_s$). Au cours de celle-ci, quand l'utilisateur se tait pendant une période supérieure à δ_s (on parle de *micro-silence*), le système prend en compte le nouveau morceau de numéro et le répète à l'utilisateur jusqu'à ce qu'après un *feedback*, l'utilisateur manifeste un silence d'une durée supérieure à Δ_s (indiqué par l'envoi de la chaîne 'longSilence'). Le client détermine les durées du silence à l'aide du nombre de micro-tours consécutifs durant lesquels on ne détecte aucune sortie de l'ASR. Alternativement, nous aurions pu utiliser la VAD (Voice Activity Detection) comme dans (Breslin *et al.*, 2013).

2. http ://dialogue.orange-labs.fr/CFAsTIncr/
3. http ://dialogue.orange-labs.fr/DictaNum/

Pendant la dictée, à chaque micro-tour utilisateur ($EndMicroTurnCond$ = horloge de fréquence 2 Hz), le *Scheduler* vérifie que la requête du client est différente de la précédente ($ServiceReqCond$ = modification de la requête) auquel cas il effectue un *rollback* puis envoie une requête au service depuis le début du morceau de numéro en cours. Quand le client détecte un micro-silence, il envoie la chaîne de caractère 'silence' au *Scheduler* (non concaténée au fragment de numéro en cours) et c'est comme cela que $CommitCond$ est définie (un *commit* est donc effectué). Par ailleurs, le client requête la dernière réponse disponible dans le *Scheduler* à l'aide d'un processus indépendant, et ce, à la même fréquence que celle des micro-tours utilisateur. Ainsi, le micro-silence étant de l'ordre de quelques micro-tours, au moment où $CommitCond$ est vérifiée, le client dispose déjà du *feedback* du service et peut le jouer immédiatement.

Faire varier δ_s entre une petite valeur (quelques micro-tours utilisateur) et Δ_s crée un continuum entre un comportement incrémental et non incrémental. Si $\delta_s = \Delta_s$, il est plus confortable pour l'utilisateur de dicter son numéro en une seule fois (situation décrite plus haut), par contre si δ_s est sensiblement plus petit que Δ_s, l'utilisateur sera interrompu dans sa dictée à chaque micro-silence pour avoir un *feedback* (il ne faut pas que δ_s soit trop petit pour que les *feedbacks* ne deviennent pas trop fréquents et donc gênants pour la dictée).

Une interaction avec la version originale du système ressemble à ceci :

– **Système :** Bienvenue dans le service DictaNum. Veuillez dicter votre numéro.
– **Utilisateur :** 01 45 65 79 98.
– **Système :** Le numéro dicté est : 01 45 67 79 98. Est-ce bien cela ?
– **Utilisateur :** Non.
– **Système :** Désolé. Nous allons reprendre du début. Veuillez dicter votre numéro.
– **Utilisateur :** 01 45 65 79 98.
– **Système :** Le numéro dicté est : 01 47 65 79 98. Est-ce bien cela ?
– **Utilisateur :** Non.
– **Système :** Désolé. Nous allons reprendre du début. Veuillez dicter votre numéro.
– [...]
– **Système :** Le numéro dicté est : 01 45 65 79 98. Est-ce bien cela ?
– **Utilisateur :** Oui.
– **Système :** Merci d'avoir utilisé DictaNum. À bientôt.

En lui intégrant un comportement incrémental, le dialogue se déroule comme suit :

– **Système :** Bienvenue dans le service DictaNum. Veuillez dicter votre numéro.
– **Utilisateur :** 01 45 (silence de durée δ_s)
– **Système :** 01 45
– **Utilisateur :** 65 79 (silence de durée δ_s)
– **Système :** 67 79
– **Utilisateur :** Non, 65 79 (silence de durée δ_s)
– **Système :** Désolé, 65 79
– **Utilisateur :** 98 (silence de durée δ_s)
– **Système :** 98
– **Utilisateur :** ... (silence de durée Δ_s)
– **Système :** Le numéro dicté est : 01 45 65 79 98. Est-ce bien cela ?
– **Utilisateur :** Oui.
– **Système :** Merci d'avoir utilisé DictaNum. À bientôt.

La Figure 4 illustre les opérations effectuées par le client et le *Scheduler* durant le dernier dialogue. Par ailleurs, il est possible d'interrompre le système durant le *feedback* final. Pour ce faire, le service renvoie le message de *feedback* sous la forme : *Le numéro dicté est : 01 <sep> 45 <sep> 65 <sep> 79 <sep> 98. Est-ce bien cela ?*. La balise *<sep>* joue le rôle de séparateur entre les différents fragments qui sont envoyés à la suite à la TTS. Une dictée peut se terminer ainsi :

– **Système :** Le numéro dicté est : 01 45 67...
– **Utilisateur :** Non, 65.
– **Système :** Pardon. Le numéro dicté est : 01 45 65 79 98. Est-ce bien cela ?
– **Utilisateur :** Oui.
– **Système :** Merci d'avoir utilisé DictaNum. À bientôt.

Tour	Micro-tour utilisateur	Action client	Action Scheduler	Commentaire
T^1	$\mu T_1^{1,U}$	requête initiale	requête initiale	Requête initiale
	$\mu T_2^{1,U}$	envoi(‘01’)	envoi(‘01’)	Obtention réponse à ‘01’
	$\mu T_3^{1,U}$	envoi(‘01 45’)	*rollback* + envoi(‘01 45’)	Obtention réponse à ‘01 45’
	$\mu T_4^{1,U}$	envoi(‘silence’)	*commit*	*Commit*
				Réponse ‘01 45’
T^2	$\mu T_1^{2,U}$	envoi(‘67’)	envoi(‘67’)	Obtention réponse à ‘67’
	$\mu T_2^{2,U}$	envoi(‘67 79’)	*rollback* + envoi(‘67 79’)	Obtention réponse à ‘67 79’
	$\mu T_3^{2,U}$	envoi(‘silence’)	*commit*	*Commit*
				Réponse ‘67 79’
T^3	$\mu T_1^{3,U}$	envoi(‘Non’)	envoi(‘Non’)	Obtention réponse à ‘Non’ (Ici, ‘Désolé’)
	$\mu T_2^{3,U}$	envoi(‘Non, 65’)	*rollback* + envoi(‘Non, 65’)	Obtention réponse à ‘Non, 65’
	$\mu T_3^{3,U}$	envoi(‘Non, 65 79’)	*rollback* + envoi(‘Non, 65 79’)	Obtention réponse à ‘Non, 65 79’
	$\mu T_4^{3,U}$	envoi(‘silence’)	*commit*	Correction et *commit*
				Réponse ‘Désolé, 65 79’
T^4	$\mu T_1^{4,U}$	envoi(‘98’)	envoi(‘98’)	Obtention réponse à ‘98’
	$\mu T_2^{4,U}$	envoi(‘silence’)	*commit*	*Commit*
				Réponse ‘98’
T^5	$\mu T_1^{5,U}$	envoi(‘longSilence’)	envoi(‘longSilence’)	Détection fin numéro
				Demande confirmation numéro
T^6	$\mu T_1^{6,U}$	envoi(‘Oui’)	envoi(‘Oui’)	Obtention réponse à ‘Oui’
	$\mu T_2^{6,U}$	envoi(‘silence’)	*commit*	*Commit*
				Message fin

FIGURE 4 – Actions effectuées par le client et le *Scheduler* dans l’exemple d’interaction avec DictaNum

Au moment de l’interruption, le message envoyé au service se présente sous la forme {*texte déjà énoncée par la TTS* | *contenu du barge-in*}. Dans le cas du dernier exemple, le service reçoit {*Le numéro dicté est : 01 45 67* | *Non, 65*} ce qui lui permet d’effectuer la correction (ou pas si l’interruption n’est qu’une confirmation par exemple).

5 Discussion

La Figure 5 présente une analyse des fonctionnalités accessibles suivant le degré d’intégration du caractère incrémental dans un système. La classification présentée en Section 2 est reprise ici. Les fonctionnalités abordées concernent les systèmes de dialogue incrémentaux et donc les systèmes de la première catégorie ne proposent aucune d’elles. À l’opposé, ces fonctionnalités ont déjà été implémentées dans des systèmes appartenant au dernier groupe.

La stratégie *List of Availabilities* de NASTIA (El Asri *et al.*, 2014) a recours à deux fonctionnalités liées du système : la possibilité de s’interrompre après avoir analysé l’entrée et le fait de pouvoir relier l’instant d’interruption de l’utilisateur au signal sonore joué par la TTS. La capacité à détecter les interruptions de l’utilisateur existe déjà dans certains systèmes traditionnels (Laroche *et al.*, 2011) mais le moment et le sens de l’intervention ne sont pas pas mis en relation avec l’énoncé du système. De plus, celui-ci ne peut pas choisir de continuer à parler en ignorant certaines interruptions. En revanche, NASTIA est conçu pour effectuer une énumération que l’utilisateur peut interrompre puis réagir en fonction du contenu de son intervention et du moment auquel elle est faite. L’utilisateur communique avec le service à l’aide d’une plate-forme vocale qui interrompt la synthèse vocale si l’utilisateur commence à parler. Pendant que la TTS est en train de prononcer une alternative, l’utilisateur peut se taire ou intervenir. Dans le premier cas, le système est relancé automatiquement pour fournir l’alternative suivante car son timeout (temps au bout duquel le système est relancé en absence de réponse de l’utilisateur) a été réglé à une valeur très proche du temps mis pour énoncer une alternative. Dans le second cas, la TTS s’interrompt et le résultat de l’ASR ainsi que le moment de l’intervention sont transmis au service qui se charge de les analyser et de les traiter. Par conséquent, deux fonctionnalités propres aux systèmes incrémentaux peuvent être reproduites dans un système de dialogue non incrémental à condition qu’il reste à l’écoute de l’utilisateur (même si la TTS est en train de jouer un message) et qu’il capte le contenu de l’intervention et le moment auquel elle a été faite. Ces conditions sont vérifiées pour les systèmes de catégorie 3 ce qui leur donne également accès à ces deux fonctionnalités.

Fonctionnalité	Catégorie 1	Catégorie 2	Catégorie 3	Catégorie 4
Le système décide de s'interrompre après analyse de l'entrée	-	+	+	+
Relier l'instant d'interruption à la TTS	-	+	+	+
Le système interrompt l'utilisateur	-	-	+	+
Réactivité améliorée	-	-	+	+
Coût de traitement optimisé	-	-	-	+

FIGURE 5 – Analyse des fonctionnalités accessibles suivant le degré d'intégration du comportement incrémental

À l'inverse, il peut être intéressant pour le système d'interrompre l'utilisateur. Par exemple, en cas de désalignement, cela ne sert à rien de laisser l'utilisateur finir sa requête et le système peut directement lui demander de la reformuler. Les *feedbacks* générés par DictaNum constituent un autre cas d'utilisation, bien que les interruptions soient moins marquées car elles interviennent après un micro-silence. Cependant, dans un système utilisant un *Scheduler*, rien n'empêche le service d'intervenir de manière spontanée à n'importe quel micro-tour utilisateur. En revanche, les systèmes de la catégorie 2 ne permettent de soumettre une requête au service qu'à la fin de l'énoncé de l'utilisateur. Ils ne peuvent donc pas interrompre l'utilisateur. Notons qu'après une interruption du système, il faut arrêter d'écouter l'utilisateur pendant quelques micro-tours afin de ne pas considérer la fin de sa phrase (le temps qu'il se rende compte que le système l'a interrompu) comme un *barge-in* de sa part.

Un des avantages majeurs du dialogue incrémental réside dans l'amélioration de la réactivité du système. Les systèmes de type 1 et 2 attendent la fin de la requête de l'utilisateur (marquée par un silence) avant de la traiter. En revanche les systèmes des catégories 3 et 4 traitent chaque nouvel incrément d'information dès son arrivée, ainsi, avant la détection du silence, la réponse du système est déjà disponible. Cela permet de l'envoyer au client à l'instant même où le seuil de silence est dépassé (retour chariot pour les services textuels). Néanmoins, les systèmes du groupe 3 ont un fonctionnement du type *restart incremental* et de ce fait, à chaque nouvel incrément d'information injecté par l'utilisateur, la requête (incomplète) est traitée intégralement depuis le début après avoir effectué un *rollback*. En revanche, les architectures de type 4 traitent les nouveaux incréments comme compléments des informations déjà disponibles. Cela permet de n'effectuer que le traitement nécessaire pour intégrer le nouvel incrément à son arrivée, optimisant ainsi les coûts de traitement. Remarquons au passage qu'une nouvelle unité incrémentale peut modifier une partie ou toute la requête, néanmoins, les systèmes du groupe 4 sont capables de gérer une telle situation en annulant les traitement qui ne sont plus en accord avec la nouvelle requête. Ce mécanisme est désigné par le terme *revoke* dans (Schlangen & Skantze, 2011). Enfin, sur nos deux exemples d'implémentation, les délais de réponse du service sont très courts. Par conséquent, il n'est pas utile d'optimiser les délais de traitement si le service répond rapidement et qu'il ne doit pas effectuer de tâches qui créent un délai (accès lent à des bases de données par exemple).

Notre solution permet d'éviter de modifier le service de dialogue au niveau fonctionnel (mis à part l'ajout du contexte de simulation). Cependant, comme c'est le cas pour DictaNum, certaines modifications au niveau applicatif peuvent être indispensables. Le rôle du *Scheduler* n'est pas de générer lui même des messages ou de faire de la gestion de dialogue au sens classique. Ainsi, quand le passage en incrémental nécessite de nouveaux types de messages tels que les *feedbacks* à l'échelle du micro-tour, ou encore la prise en compte des corrections (*Non, 65...*), il faut les implémenter dans le service.

Enfin, afin de pouvoir prendre des décisions de prise de parole optimales, le *Scheduler* pourrait avoir besoin de certaines informations provenant de modules plus en aval dans la chaîne de dialogue (au niveau du service). Encore une fois, cela doit être géré au niveau applicatif. Un futur papier, dédié à l'implémentation des systèmes de dialogue en utilisant le *Scheduler*, traitera les idées brièvement décrites dans les deux derniers paragraphes.

6 Conclusion et travaux futurs

En partant d'un système de dialogue non-incrémental, ce papier montre qu'il est possible d'en rendre le fonctionnement incrémental avec peu de modifications. Le *Scheduler*, un module intermédiaire entre le client et le service de dialogue permet de simuler un comportement incrémental vu du client tout en laissant le service pratiquement inchangé. Au fur et à mesure que l'utilisateur parle, des incréments de requête sont formés et le *Scheduler* détermine les réponses correspondantes en les envoyant au service. Au bout d'un moment, le *Scheduler* prend la décision d'engager le dialogue dans le sens de la dernière hypothèse de requête en cours en envoyant un signal de *commit* au système. Ce dernier peut ainsi se baser sur cette hypothèse pour agir sur le monde extérieur (formulation d'une réponse, modification d'une base de données...).

Traditionnellement, le service maintient un contexte de dialogue constitué de toutes les informations relatives au déroulement du dialogue actuel. Dans le cadre de notre solution, il faudra maintenir un second contexte qui sera modifié lors des requêtes non validées par un *commit*. La plupart des requêtes adressées au service servent à voir ce qu'il aurait répondu dans le cas où l'hypothèse actuelle est la bonne et on ne veut pas que ces requêtes modifient le contexte principal.

Comme application de la solution proposée ici, deux services de dialogue déjà existants ont été transformés en systèmes incrémentaux. Le premier est un service en mode texte dont l'objectif est d'aider l'utilisateur à naviguer dans une base de contenus (la FAQ de l'ANPE en ce qui nous concerne). Le second, en mode vocal, invite l'utilisateur à dicter un numéro.

Concevoir un système de dialogue incrémental capable de décider quand réaliser un *commit* et prendre la parole n'est pas facile à concevoir de manière experte. Par la suite, nous envisageons d'explorer des approches basées sur l'apprentissage statistique pour optimiser en ligne ces comportements et ainsi atteindre cet objectif.

Références

ALLEN J., FERGUSON G. & STENT A. (2001). An architecture for more realistic conversational systems. In *6th international conference on Intelligent user interfaces*.

BAUMANN T. & SCHLANGEN D. (2013). Open-ended, extensible system utterances are preferred, even if they require filled pauses. In *Proceedings of the SIGDIAL 2013 Conference*.

BRESLIN C., GASIC M., HENDERSON M., KIM D., SZUMMER M., THOMSON B., TSIAKOULIS P. & YOUNG S. (2013). Continuous asr for flexible incremental dialogue. In *ICASSP*, p. 8362–8366.

BROWN-SCHMIDT S. & HANNA J. E. (2011). Talking in another person's shoes : Incremental perspective-taking in language processing. *Dialogue and Discourse*, **2**, 11–33.

CLARK H. H. (1996). *Using Language*. Cambridge University Press.

DEVAULT D., SAGAE K. & TRAUM D. (2011). Incremental interpretation and prediction of utterance meaning for interactive dialogue. *Dialogue and Discourse*, **2**, 143–170.

DOHSAKA K. & SHIMAZU A. (1997). A system architecture for spoken utterance production in collaborative dialogue. In *IJCAI*.

EDLUND J., GUSTAFSON J., HELDNER M. & HJALMARSSON A. (2008). Towards human-like spoken dialogue systems. *Speech Communication*, **50**, 630–645.

EL ASRI L., LEMONNIER R., LAROCHE R., PIETQUIN O. & KHOUZAIMI H. (2014). NASTIA : Negotiating Appointment Setting Interface. In *Proceedings of LREC*.

FINK G. A., SCHILLO C., KUMMERT F. & SAGERER G. (1998). Incremental speech recognition for multimodal interfaces. In *IECON*.

HASTIE H., AUFAURE M.-A. *et al.* (2013). Demonstration of the parlance system : a data-driven incremental, spoken dialogue system for interactive search. In *Proceedings of the SIGDIAL 2013 Conference*.

ILKIN Z. & STURT P. (2011). Active prediction of syntactic information during sentence processing. *Dialogue and Discourse*, **2**, 35–58.

LANGKILDE I., WALKER M. A., WRIGHT J., GORIN A. & LITMAN D. (1999). Automatic prediction of problematic human-computer dialogues in how may i help you ? In *ASRU99*.

LAROCHE R. (2010). *Raisonnement sur les incertitudes et apprentissage pour les systemes de dialogue conventionnels*. PhD thesis, Paris VI University.

LAROCHE R. & PUTOIS G. (2010). *D5.5 : Advanced Appointment-Scheduling System "System 4"*. Prototype D5.5, CLASSIC Project.

LAROCHE R., PUTOIS G. *et al.* (2011). *D6.4 : Final evaluation of CLASSiC TownInfo and Appointment Scheduling systems*. Report D6.4, CLASSIC Project.

LEVELT W. J. M. (1989). *Speaking : From Intention to Articulation*. Cambridge, MA : MIT Press.

LOCK K. (1965). Structuring programs for multiprogram time-sharing on-line applications. In *AFIPS '65 (Fall, part I) Proceedings of the November 30–December 1, 1965, fall joint computer conference, part I*.

MATSUYAMA K., KOMATANI K., OGATA T. & OKUNO H. G. (2009). Enabling a user to specify an item at any time during system enumeration – item identification for barge-in-able conversational dialogue systems –. In *Proceedings of the INTERSPEECH 2009 Conference*.

MATTHIAS G. M. (2008). Incremental speech understanding in a multimodal web-based spoken dialogue system. Master's thesis, Massachusetts Institute of Technology.

RAUX A. & ESKENAZI M. (2008). Optimizing endpointing thresholds using dialogue features in a spoken dialogue system. In *SIGDIAL*.

SCHLANGEN D. & SKANTZE G. (2011). A general, abstract model of incremental dialogue processing. *Dialogue and Discourse*, **2**, 83–111.

SELFRIDGE E., ARIZMENDI I., HEEMAN P. & WILLIAMS J. (2013). Continuously predicting and processing barge-in during a live spoken dialogue task. In *Proceedings of the SIGDIAL 2013 Conference*.

SELFRIDGE E. O., ARIZMENDI I., HEEMAN P. A. & WILLIAMS J. D. (2012). Integrating incremental speech recognition and pomdp-based dialogue systems. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*.

SKANTZE G. & SCHLANGEN D. (2009). Incremental dialogue processing in a micro-domain. In *ACL*.

TANENHAUS M. K., SPIVEY-KNOWLTON M. J., EBERHARD K. M. & SEDIVY J. C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, **268**, 1632–1634.

WIRÉN M. (1992). *Studies in Incremental Natural Language Analysis*. PhD thesis, Linköping University, Linköping, Sweden.

WLODARCZAK M. & WAGNER P. (2013). Effects of talk-spurt silence boundary thresholds on distribution of gaps and overlaps. In *INTERSPEECH Proceedings*.

La base lexicale Démonette : entre sémantique constructionnelle et morphologie dérivationnelle

Nabil Hathout[1] , Fiammetta Namer[2]
(1) UMR 5263 CLLE/ERSS, CNRS & Université Toulouse Le Mirail, Toulouse
(2) UMR 7118 ATILF, CNRS & Université de Lorraine, Nancy
Nabil.Hathout@univ-tlse2.fr, Fiammetta.Namer@univ-lorraine.fr

Résumé. Démonette est une base de données lexicale pour le français dont les sommets (entrées lexicales) et les arcs (relations morphologiques entre les sommets) sont annotés au moyen d'informations morpho-sémantiques. Elle résulte d'une conception originale intégrant deux approches radicalement opposées : Morphonette, une ressource basée sur les analogies dérivationnelles, et DériF, un analyseur à base de règles linguistiques. Pour autant, Démonette n'est pas la simple fusion de deux ressources pré-existantes : cette base possède une architecture compatible avec l'approche lexématique de la morphologie ; son contenu peut être étendu au moyen de données issues de sources diverses. L'article présente le modèle Démonette et le contenu de sa version actuelle : 31 204 verbes, noms d'action, noms d'agent, et adjectifs de propriété dont les liens morphologiques donnent à voir des définitions bi-orientées entre ascendants et entre lexèmes en relation indirecte. Nous proposons enfin une évaluation de Démonette qui comparée à Verbaction obtient un score de 84% en rappel et de 90% en précision.

Abstract. Démonette is a lexical database whose vertices (lexical entries) and edges (morphological relations between the vertices) are annotated with morpho-semantic information. It results from an original design incorporating two radically different approaches: Morphonette, a resource based on derivational analogies and DériF, an analyzer based on linguistic rules. However, Daemonette is not a simple merger of two pre-existing ressources: its architecture is fully compatible with the lexematic approach to morphology; its contents can be extended using data from various other sources. The article presents the Démonette model and the content of its current version, including 31,204 verbs, action nouns, agent nouns and property adjectives, where morphological links between both direct ascendants and indirectly related words have bi-oriented definitions. Finally, Démonette is assessed with respect to Verbaction with a recall of 84% and a precision of 90%.

Mots-clés : Réseau lexical. Morphologie dérivationnelle. Famille morphologique. Sémantique lexicale. Français
Keywords: Lexical Network. Derivational morphology. Morphological family. Lexical semantics. French.

1 Introduction

Cet article présente le réseau morpho-sémantique Démonette. Cette ressource comporte plus de 30 000 noms, verbes et adjectifs, dont les connections réalisent les relations morphologiques directes ou indirectes entre mots de la même famille dérivationnelle[1]. Les éléments de ce réseau sont des lexèmes typés sémantiquement ; les arcs sont valués par les relations morpho-sémantiques qui s'établissent entre ces mots ; ces relations sont regroupées en fonctions de leurs définitions abstraites. Ce réseau est le fruit d'un travail qui s'inscrit dans la continuité d'une étude préliminaire présentée dans (Hathout, Namer, 2011) qui montrait la complémentarité de deux ressources morphologiques dérivationnelles conçues selon des principes théoriques opposés, DériF et Morphonette, et comment elles peuvent être combinées.

La suite de l'article est organisée comme suit. §2 présente le modèle de Démonette, et notamment la conception cumulative du sens lexical construit (§2.2), le typage de ces mots construits, ainsi que les gloses[2] et leur abstraction

[1] Une famille dérivationnelle est un ensemble de mots apparentés directement ou indirectement. Elle comporte généralement un mot simple (morphologiquement indécomposable) et plusieurs mots construits. Les mots construits se répartissent entre composés (POISSON-CHAT ← POISSON, CHAT) et dérivés par suffixation (CHANT**EUR** ← CHANTER), par préfixation (**IM**PUR ← PUR) ou par conversion (SCIER ← SCIE). Dans cet article, il ne sera question que de dérivation.

[2] Une glose est la paraphrase en langue naturelle du mot construit par rapport au parent morphologique (i.e. au membre de la famille dérivationnelle) auquel il est relié.

(§2.3). Le §3 détaille la construction proprement dite du réseau lexical. Le §3.1 décrit le corpus de travail. Les §3.2 et 3.3 présentent respectivement DériF et Morphonette ainsi que la manière dont nous en avons extrait les informations destinées à alimenter Démonette. Une caractérisation qualitative et quantitative de ce lexique est proposée au §3.4. Suit en §4 une évaluation partielle de Démonette relativement au lexique Verbaction[3]. Nous abordons ensuite en §5 différentes questions liées à l'extension de cette nouvelle ressource et à la possibilité de découvrir et de caractériser des relations dérivationnelles qui composent Démonette sans passer par DériF ni Morphonette.

2 Le modèle Démonette

2.1 Une nouvelle ressource :justification

Un très grand nombre d'outils d'analyse morphologique, tant flexionnelle que constructionnelle ont été développés au cours des deux dernières décennies. Les travaux récents les plus visibles ont plutôt été réalisés par les informaticiens pour répondre à des besoins de décomposition morphologique et pallier les limites des lexiques. Leur objectif est le plus souvent le découpage des mots en morphèmes. Certains systèmes opèrent en plus un étiquetage des morphèmes permettant de regrouper leurs variantes. Ces systèmes utilisent tous des méthodes d'apprentissage automatique. Ils sont relativement indépendants des langues, mais ont tendance à produire de meilleurs résultats pour les langues à morphologie concaténative comme l'anglais, le français ou l'allemand. Citons notamment Linguistica (Goldsmith, 2001), Morphessor (Creutz, Lagus, 2005) ou l'analyseur de Bernhard (2006). Plusieurs de ces travaux ont été développés autour des campagnes Morpho-Challenge dont l'un des enseignements est que, contrairement à d'autres tâches comme l'annotation morpho-syntaxique, les performances des analyseurs morphologiques statistiques est globalement trop faible pour améliorer significativement les applications dans lesquelles ils seraient intégrés. On trouve à côté de ces systèmes statistiques des analyseurs plus « linguistiques » comme DériF (Namer, 2009) ; pour un panorama général, voir par exemple (Bernhard et al., 2011 ; Namer, 2013).

Parallèlement, on observe qu'aucun lexique dérivationnel de taille suffisante n'a été développé dans les 20 dernières années et que, depuis la création et la diffusion de CELEX (Baayen et al., 1995), aucune ressource similaire n'a été réalisée ou mise à disposition des linguistes, psycholinguistes et linguistes informaticiens. Ce manque est surtout crucial pour les langues romanes, notamment le français, l'italien et l'espagnol. Le constat était déjà fait il y a plus de 15 ans par Dal et al. (1999). Il reste inchangé.

Certaines ressources apportent certes des réponses partielles à ce besoin. Le réseau lexical JeuxDeMots (Lafourcade, Joubert, 2013) contient des relations morphologiques regroupées en classes dont certaines sont typées sémantiquement au moyen d'étiquettes comme 'verbe-action', 'action-verbe', etc. Cette ressource, constituée par les participants à plusieurs jeux en ligne bénéficie d'une couverture lexicale très large mais présente un défaut de systématicité dans les relations lexicales codées et notamment dans les relations morphologiques. Des relations morpho-sémantiques sont également présentes dans les réseaux WordNet de plusieurs langues comme l'anglais (Fellbaum, Miller, 2003), le tchèque (Pala, Hlaváčková, 2007) ou le turc (Bilgin et al., 2004). Dans le Princeton WordNet 3.0, par exemple, les relations morphologiques ne sont pas sémantiquement typées, et ne connectent que certains synsets (dont tous les mots ne sont pas concernés). On trouve par ailleurs au niveau des synsets des gloses du sens construit morphologiquement, mais qui ne mettent pas formellement en jeu le sens du synset de la base. Ces relations sont en revanche présentes dans le réseau lexical FrameNet (Ruppenhofer et el., 2002) qui comporte des gloses « abstraites » similaires à celle de Démonette (cf *infra*). Comme nous allons le voir, ces deux ressources partagent par ailleurs une organisation du sens lexical en « constellation » centrée autour des prédicats verbaux.

À terme, l'apport de Démonette sera triple. Il permettra d'une part d'offrir des **analyses morphologiques fiables** pour une langue qui ne dispose toujours pas d'un véritable lexique dérivationnel : la fiabilité est l'un des points forts de DériF (cf §3.1) dont Démonette bénéficie directement. La deuxième contribution de Démonette sera la construction d'une **ressource lexicale dérivationnelle** de grande taille[4]. La troisième est une structure lexicale en réseau qui coïncide parfaitement avec l'**approche lexématique** de la morphologie lexicale moderne (pour le français, cf entre autres Fradin, 2003 ; Roché et al., 2011). L'architecture de Démonette illustre en effet comment sémantique et morphologie constructionnelle peuvent s'articuler dans ce type d'approche.

2.2 Réseau morphologique dérivationnel

Démonette présente une architecture relativement originale. C'est un réseau lexical implémenté sous la forme d'un graphe où chaque mot est connecté à une partie des membres de sa famille dérivationnelle. Les mots construits sont

[3] http://redac.univ-tlse2.fr/lexiques/verbaction.html

[4] Dans sa version actuelle, Démonette inclut environ 15% de la nomenclature d'un dictionnaire comme le *Trésor de la Langue Française* (TLF).

reliés à leur base (par des **relations « directes »**) et à un sous-ensemble des autres mots construits de leur famille dérivationnelle (par des **relations « indirectes »**). Cette organisation reprend en partie celle de Morphonette (§3.2). Par exemple, les arcs en (1) relient les membres la famille de *essorer*. Les relations « directes » comme *essorage* – *essorer* sont notée en gras et les relations « indirecte » comme *essorage* – *essoreuse* en maigre.

(1) **essorage – essorer** ; essorage – essoreur ; essorage – essoreuse ;
essoreur – essorer ; essoreur – essorage ; essoreur – essoreuse ;
essoreuse – essorer ; etc.

Les sommets qui représentent les mots contiennent des informations sur la catégorie, les traits morphosyntaxiques et éventuellement, le type d'affixation ayant permis de les construire. Par exemple, le sommet qui représente *essorage* indique qu'il s'agit d'un nom masculin construit par une suffixation en *-age*. Démonette offre ainsi une représentation morphologique complète des mots qu'elle décrit. Il fournit en outre un typage sémantique de ces mots et une description de leur **sens morphologique**.

2.3 Sémantique cumulative

L'objectif de Démonette est à terme de fournir pour chaque couple de mots du TLF, reliés directement ou indirectement, une annotation bi-orientée, de manière à ce que chaque mot construit soit défini relativement à chacun des mots avec lesquels il entretient une parenté morphologique. Cet objectif est plus facilement atteignable quand la relation est directe, c'est-à-dire entre un dérivé et sa base, grâce notamment aux gloses du sens du dérivé fournies par DériF (cf §3.1). En revanche le traitement des relations indirectes réclame davantage de réflexion (voir §5). Si on considère par exemple la famille partielle donnée en (2), l'étiquetage sémantique de l'arc entre *activateur* et *activation* est justifié (d'autant qu'il est reconstitué spontanément par les locuteurs, qui voient en *activateur* le nom désignant l'auteur d'une *activation*, et, réciproquement, dans le nom *activation*, l'acte réalisé par *l'activateur*) alors qu'il semble moins légitime entre le nom *activateur* et l'adverbe *activement*.

(2) (actif, activer, activateur, activatrice, activation, activiste, activisme, activement, …)

2.3.1 Des descriptions sémantiques redondantes

Chaque relation morphologique apporte une contribution au sens des mots qu'elle connecte. Ces sens élémentaires se cumulent alors pour produire le sens global. On peut considérer que le sens global d'une relation entre deux parents, e.g. *momifiable* et *momie* (3c) se déduit de la composition des relations élémentaires que chaque parent entretient avec son ascendant direct, ici *momifiable* et *momifier* d'une part (3a) et *momifier* et *momie* de l'autre (3b) :

(3) a. momifiable/ADJ ← momifier/VERBE : « que l'on peut momifier »
b. momifier/VERBE ← momie/NOM : « transformer en momie »
c. momifiable/ADJ ← momie/NOM : « que l'on peut transformer en momie »

De même, on peut généralement calculer, l'un relativement à l'autre, le sens de deux mots complexes qui ont la même base : *momifiable* peut être défini relativement à *momifier* (3a) et *momifier* est connecté à *momification* par une relation que l'on peut gloser comme en (4a). En conséquence, *momifiable* peut tirer son sens de *momification* (4b), et réciproquement (4c).

(4) a. momification/NOM ← momifier/VERBE « action de momifier »
b. momifiable/ADJ ← momification/NOM : « à qui il est possible d'appliquer la momification »
c. momification/NOM ← momifiable/ADJ : « acte applicable à ce(lui) qui est momifiable »

Les sens élémentaires des mots complexes apparentés sont ainsi interconnectés par le partage de prédicats et d'arguments. C'est cette faculté qui est exploitée dans Démonette : chaque mot y reçoit autant de définitions qu'il a de connections avec ses parents morphologiques. Dans sa version présentée ici, les descriptions sémantiques de Démonette sont exprimées en langage semi-naturel dans le but de rendre plus explicites ces interconnexions.

2.3.2 Séries dérivationnelles et typage sémantique des unités lexicales

Les mots et les relations sémantiques qui les connectent peuvent être regroupés en classes qui correspondent à des types sémantiques. Cette classification repose sur un niveau d'abstraction qui met en jeu des variables typées illustrées dans la table 1 et qui nous permet de comparer entre elles les unités et leur sens. Dans les représentations abstraites, le symbole @ correspond au prédicat verbal à partir duquel se définissent les noms et adjectifs qui lui sont apparentés ; de même, on note @FCT le nom ou l'adjectif qui remplit la fonction sémantique FCT vis-à-vis du prédicat @.

Variable typée	Signification	Séries dérivationnelles	Exemples
@ACT	Nom d'activité	Xment, Xage, Xion	*lavage, gonflement, exclusion*
@AGM	Nom d'agent masculin	Xeur	*danseur, sauveteur*
@AGF	Nom d'agent féminin	Xeuse, Xrice	*danseuse, sculptrice*
@PROP	Adjectif de propriété	Xif	*combatif*
@	Prédicat	–	*combattre, laver, danser, sculpter, gonfler, exclure*

TABLE 1 : typage sémantique des unités lexicales de Démonette

Par exemple, le dérivé *danseur* dénote l'(AG)ent (M)asculin du prédicat @ réalisé par le verbe d'action *danser*. Son étiquette est @AGM. Plus généralement, le type sémantique d'un mot *Y* appartenant à une famille dérivationnelle constituée autour d'un prédicat *X* est déterminée par la **série dérivationnelle** de *Y,* c'est-à-dire l'ensemble des mots construits par la même règle de dérivation que *Y*. Cette série dérivationnelle fournit ainsi l'identifiant du rôle sémantique que *Y* exerce vis-à-vis de *X*.

Actuellement, Démonette contient les membres de 7 séries dérivationnelles et leurs bases. Nous y trouvons une partie des verbes qui dénotent des actions, les séries dérivationnelles déverbales des noms d'**action** en *-age, -ment* et *-ion* (dénotant les mêmes actions que leur verbe de base) et des noms **d'agent** en *-eur, -euse* et *-rice*, (référent à des humains susceptibles de réaliser ces actions) et les adjectifs de **propriété** en *-if* (liés aux potentialités d'actualisation du prédicat verbal). Tous ces modes de formation sont facilement identifiables, productifs, et sémantiquement réguliers. Ils sont traités par les deux systèmes qui alimentent Démonette. De plus, les parentés indirectes y sont nombreuses et induisent des connections transverses intéressantes, cf (5), faciles à gloser, comme en (6) pour la relation entre *ravitailleur* et *ravitaillement.*

(5) a. ravitailler, ravitaillement, ravitailleur, ravitailleuse
b. capter, captage, captation, captatif, captateur, capteur
c. décanter, décantage, décantation, décantement, décanteur

(6) *ravitailleur*: (Agent masculin habituel - Auteur masculin exceptionnel - Instrument) de l'activité liée au *ravitaillement*

Dans certaines familles comme en (7), le verbe est absent. Les radicaux savants /predat/ et /odit/, communs aux membres de ces familles respectives sont la trace laissée par un prédicat en latin qui n'a pas d'héritier verbal en français contemporain. Néanmoins, les familles (7a) et (7b) sont composées de membres « légitimes » des séries dérivationnelles que nous avons sélectionnées, étant donné les propriétés formelles de ces mots et les caractéristiques sémantiques qui leur sont corrélées (*audition* dénote une action, *prédateur*, une classe d'agents, *auditif*, une propriété, etc.).

(7) a. prédateur, prédation, prédatrice
b. audition, auditif, auditeur, auditrice

2.3.3 Annotations sémantiques des relations entre unités lexicales

Les types sémantiques présentés ci-dessus sont utilisés comme paramètres dans les définitions associées aux relations entre les dérivés et leurs bases (relations directes, comme *fonder* → *fondateur*) ou entre des couples de mots partageant un même ascendant (relations indirectes, comme *fondateur* → *fondation*). Les **définitions abstraites** associées aux relations directes et indirectes sont établies selon les trois principes suivants :

1. Chaque membre d'une relation est représenté par son **type sémantique**.
2. Dans une relation directe entre un dérivé et sa base, le dérivé est défini par rapport au sens de sa base (définition dite **orientée**). Réciproquement, on indique par une définition **orientée inverse** la paraphrase du sens de la base relativement à celui du dérivé.
3. Une relation indirecte connecte deux mots complexes. Chacun y est défini par rapport à l'autre, donnant lieu à deux définitions **transversales**, symétriques l'une de l'autre, qui exploitent la possibilité de combiner les sens élémentaires de ces deux mots.

Ces principes permettent d'annoter uniformément —i.e. au moyen de deux définitions symétriques— l'ensemble des relations qui existent entre les couples de mots morphologiquement apparentées dans le graphe. Les définitions orientées, extraites des gloses générées par DériF (cf §3.1), servent à l'élaboration des autres annotations, orientées inverses et transversales. L'application des trois principes est illustrée dans les tables 2 et 3. La formulation de la

définition en colonne 2 fait systématiquement intervenir le prédicat @ ; le symbole 'OU' rend compte de la polysémie de ces mots construits. Notons que les relations indirectes peuvent s'établir entre des mots qui portent la même étiquette sémantique comme les dérivés concurrents (*lavage* / *lavement* ; *ausculteur* / *auscultateur*) ou les correspondants féminins / masculins du même agent (*danseuse* / *danseur*).

Schéma de relation directe dérivé/base	**Définition orientée dérivé ← base**	**Définition orientée inverse base ← dérivé**	**Exemple**
$Xeur_N/X_V$	(Agent masculin habituel OU Auteur masculin exceptionnel OU Instrument) de @	Réaliser l'activité dont l'agent masculin est le @AGM	marcheur / marcher
$Xeuse_N/X_V$; $Xrice_N/X_V$	(Agent féminin habituel OU Auteur féminin exceptionnel OU Instrument) de @	Réaliser l'activité dont l'agent féminin est le @AGF	sculptrice / sculpter
$Xage_N/X_v$; $Xment_N/X_V$; $Xion_N/X_V$	(Action OU Résultat de l'action) de @	Réaliser le/la @ACT	abaissement / abaisser
Xif_A/X_V	En rapport avec l'acte de @	Manifester le fait d'être @PROP	combatif / combattre

TABLE 2 : Définitions orientées implémentées dans Démonette

Schéma de relation indirecte dérivé$_1$/dérivé$_2$	**Définition transversale de dérivé$_1$ ← dérivé$_2$**	**Définition transversale de dérivé$_2$ ← dérivé$_1$**	**Exemple**
($Xion_N$; $Xage_N$; $Xment_N$)/$Xeur_N$	action pratiquée par @AGM	agent masculin ou instrument du @ACT	distraction / distracteur
($Xion_N$; $Xage_N$; $Xment_N$)/ ($Xeuse_N$; $Xrice_N$)	action pratiquée par @AGF	agent féminin ou instrument du @ACT	fondation / fondatrice
Xif_A/($Xage_N$; $Xment_N$; $Xion_N$)	qui permet la @ACT	action de ce qui est @PROP	déterminatif / détermination
$Xif_A/Xeur_N$	qui caractérise l'activité pratiquée par @AGM	celui dont l'activité est @PROP	administratif / administrateur
Xif_A/($Xeuse_N$; $Xrice_N$)	qui caractérise l'activité pratiquée par @AGF	celle dont l'activité est @PROP	spoliatif / spoliatrice
($Xeuse_N$; $Xrice_N$)/$Xeur_N$	celle qui a pour correspondant masculin le @AGM	celui qui a pour correspondant féminin la @AGF	danseuse / danseur
Xage/(Xion ; Xment) ; Xment/Xion	@ACT	@ACT	ruminement / rumination
$Xeur_N$ /X'eur$_N$	@AGM	@AGM	activateur / activeur

TABLE 3 : Liste des définitions transversales implémentées dans Démonette

3 Données, méthodes et résultats

La structure générale de Démonette permet d'organiser d'une manière originale les informations morphologiques présentes dans les analyses de DériF et dans Morphonette. Pour permettre leur alignement et faciliter leur intégration, DériF a été appliqué sur le corpus à partir duquel Morphonette a été construit : TLFnome[5]. Le graphe Démonette est donc construit à partir d'un ensemble de sommets unique et connu à l'avance.

[5] L'ATILF distribue Morphalou, une version XML de TLFnome : www.cnrtl.fr/lexiques/morphalou/. TLFnome est un lexique créé à partir de la nomenclature du dictionnaire TLF. Il comporte 97 000 lemmes et se caractérise par une qualité quasi parfaite grâce aux très nombreuses révisions manuelles dont il a bénéficié.

3.1 DériF

DériF (Namer, 2009, 2013) est un analyseur qui implémente des règles de construction morphologiques établies et mises au point manuellement, par des linguistes. Il se distingue par deux caractéristiques principales :

1. Il est fiable grâce notamment à un ensemble d'exceptions permettant de prendre en compte efficacement les irrégularités qui se sont accumulées au cours de l'évolution du lexique.
2. Il propose pour chaque dérivation une glose du sens construit, similaire à celles de WordNet (cf §2.1).

DériF analyse des lemmes munis d'une catégorie grammaticale. Il met en jeu des règles établies suivant des critères linguistiques : l'analyse proposée (ou les analyses, quand plusieurs sont possibles) réunit les ascendants du mot, et la relation constructionnelle de ce dernier avec son ou ses ascendants immédiats, sous la forme d'une paraphrase, cf §2.3.1. Pour les affixations et les conversions, l'ascendant immédiat est la base comme en (8). De par sa conception, DériF est aussi capable d'analyser des néologismes. Il prédit leur base et calcule leur sens construit en fonction de leur structure morphologique et leur catégorie grammaticale (en (9), *schtroumpfement est* un nom suffixé en *-ment*).

(8) enterrement/NOM : (enterrement/NOM, enterrer/VERBE, terre/NOM) :
"(Action OU résultat de l'action) d'enterrer"

(9) schtroumpfement/NOM : (schtroumpfement/NOM, schtroumpfer/VERBE)
"(Action OU résultat de l'action) de schtroumpfer"

DériF se distingue des autres analyseurs morphologiques par sa capacité à prendre en compte la lexicalisation et notamment l'opacification de certains mots du lexique enregistré. Ce phénomène est géré au moyen de listes d'exceptions qui bloquent l'application des règles pour ces mots. Par exemple, l'inscription de *pension* parmi les exceptions évite au système de l'analyser comme déverbal de *penser* (10a), sur le modèle de *pression*, dérivé de *presser* (10b).

(10) a. pension/NOM : (pension/NOM)
b. pression/NOM : (pression/NOM, presser/VERBE)

Les analyses de DériF servent à la fois à créer la structure du graphe de Démonette —la liste des sommets et des arcs — et à en calculer les attributs. Les analyses obtenues en appliquant DériF à TLFnome permettnet de construire un premier graphe dans lequel les sommets sont les entrées de TLFnome et les arcs sont les relations dérivationnelles directes entre les mots construits et leurs ascendants directs (bases ou éléments de composition). Les sommets du graphe qui forment le corpus de Démonette sont ensuite alors en 3 temps : 1/ sélection des entrées de noms construits par l'une des 6 suffixations déverbales ; 2/ ajout des entrées des verbes de base correspondants ; 3/ ajout des entrées des adjectifs en *-if* dont la base est un nom ou un verbe préalablement sélectionné. Dans un second temps, les familles dérivationnelles sont munies de relations « indirectes ». La structure de graphe ainsi obtenue répond alors aux spécifications présentées en §2.1 et peut être directement intégrée à Démonette.

Outre la structure du graphe, les analyses de DériF permettent d'annoter les relations directes de Démonette au moyen des paraphrases (ou gloses) qui définissent les dérivés par rapport à leur base. D'autres attributs sont également déduits des analyses de DériF :

1. Les types sémantiques des sommets qui représentent des mots construits sont calculés à partir de la catégorie grammaticale (pour les verbes) ou de l'affixation utilisée pour construire le dérivé (pour les noms et adjectifs). Une unité lexicale peut théoriquement appartenir à plusieurs classes sémantico-aspectuelles correspondant à ses différentes acceptions. Dans les prochaines versions de Démonette intégrant davantage d'entrées, les mots pourront avoir plusieurs types associés à différentes relations dérivationnelles.
2. La définition abstraite de la relation directe est calculée à partir de la paraphrase concrète correspondante.
3. Les définitions abstraites transversales sont calculées en fonction des types sémantiques des sommets connectés.
4. À partir de ces définitions, ont détermine les instances concrètes des relations indirectes.

3.2 Morphonette

Morphonette est un réseau lexical du français basé sur une conception relationnelle et paradigmatique de la morphologie (Hathout, 2011). Dans ce lexique, les propriétés morphologiques d'un mot sont décrites par les paradigmes qui le contiennent. Par exemple, les propriétés d'un dérivé comme *modifiable* peuvent être minimalement décrites par la famille dérivationnelle qui rassemble les mots *modifier*, *modification*, *modificateur*, *modificatif*, *modifiant*, *modifieur, immodifiable*, etc. et par la série des dérivés en *-able* : *agaçable*, *agitable*, *chevauchable*, *définissable*, *différenciable*, *rechargeable*, *réconciliable*, *soutenable*, etc.

Morphonette a été construit à partir de TLFnome. Il est composé de **filaments**, c'est-à-dire de triplets (m, p, $s_p(m)$) où m est une entrée, p est un membre de la famille dérivationnelle de m et $s_p(m)$ est la sous-série dérivationnelle de m

relativement à p. $s_p(m)$ est l'ensemble des mots qui se trouvent dans une relation similaire à celle que m entretient avec p. En d'autres termes, un mot u appartient à $s_p(m)$ s'il existe un mot v tel que $m : p = u : v$ (i.e. tel que m, p, u, v forment une analogie). L'exemple (15) présente le filament de l'adjectif *modifiable* pour p = *modificateur*.

(15) (modifiable, modificateur,
{amplifiable, glorifiable, identifiable, justifiable, clarifiable, mystifiable, rectifiable,
sanctifiable, simplifiable, spécifiable, unifiable, vérifiable})

Dans Morphonette, une entrée a autant de filaments qu'il y a de membres dans sa famille dérivationnelle. Certains de ces filaments se recouvrent en grande partie. D'autres filaments décrivent des propriétés différentes d'un même mot. Par exemple, *travailleur* a dans sa famille le nom *travailleuse* et le verbe *travailler*. Sa sous-série relativement au verbe contient des dérivés comme *ravageur* ou *cisailleur* qui peuvent être rattachés aux verbes *ravager* ou *cisailler*. En revanche sa sous-série relative au nom féminin *travailleuse* contient en plus les noms comme *deuilleur* (← *deuil*) ou *volailleur* (← *volaille*) pour lesquels il existe un féminin en /øz/ (*deuilleuse*, *volailleuse*) mais pas de verbe correspondant. *Travailleur* appartient donc à deux sous-séries différentes, l'une correspondant à la propriété d'être un nom déverbal et l'autre à celle d'être associée à un féminin en /øz/.

L'un des intérêts des filaments est qu'ils décrivent à la fois les relations directes entre un dérivé et sa base, indirectes avec les autres membres de sa famille et ses relations avec les membres de ses sous-séries. Les relations entre mots de la même famille sont utilisées pour alimenter Démonette. Morphonette ne donnant aucune information explicite sur les opérations qui ont permis de construire les dérivés, celles-ci sont recalculées en reconstituant les couples de mots qui partagent le même radical, comme par exemple en (16), puis en réunissant ceux qui instancient la même relation de parenté morphologique, comme en (17) entre les noms en *-euse* et les adjectifs en *-if*.

(16) ('bouilleur/NOM', 'bouillage/NOM')

(17) collectif/ADJ:collectionneuse/NOM = dissertatif/ADJ:disserteuse/NOM =
nocif/ADJ:noceuse/N = perfectif/ADJ:perfectionneuse/NOM = portatif/ADJ:porteuse/NOM =
possessif/ADJ:possesseuse/NOM = sélectif/ADJ:sélectionneuse/NOM

3.3 Contenu de Démonette

Dans sa version actuelle, Démonette comporte 31 204 relations décrivant chacune un couple de mots morphologiquement apparentés. Ces couples appartiennent aux séries dérivationnelles présentées au §2.2.2, fournies soit par DériF (21 556 couples issus des familles dérivationnelles regroupant l'ensemble des ascendants des mots construits sur un prédicat verbal), soit par Morphonette (9 648 relations, toutes indirectes).

MOT1	MOT2	Or	SEM1	SEM2	REL DIR1	REL INDIR1	REL DIR2	REL INDIR2
administratif /ADJ	administratrice /NOM	D	@PROP	@AGF	En rapport avec l'acte de @	qui caractérise l'activité pratiquée par @AGF	Agent féminin ou instrument de @	celle dont l'activité est @PROP
administrateur /NOM	administrer /VERBE	D	@AGM	@	(Agent masculin habituel - Auteur masculin exceptionnel - Instrument) de @		Réaliser l'activité dont l'agent féminin est le @AGM	
admonitrice /NOM	admonition /NOM	M	@AGF	@ACT		agent féminin ou instrument du @ACT		Action pratiquée par @AGF

TABLE 4 : Échantillon de Démonette

Une entrée de Démonette comporte 9 champs (table 4) : MOT1 et MOT2 sont les membres du couple décrit ; ORIGINE renseigne sur la provenance de l'entrée (Morphonette ou DériF) ; SEM1 et SEM2 enregistrent respectivement le type sémantique de MOT1 et MOT2, RELATION INDIRECTE1 est la définition transversale de MOT1 par rapport au sens de MOT2 ; RELATION INDIRECTE2 est celle de MOT2 relativement au sens de MOT1. Enfin, si MOT1 et MOT2 entretiennent une relation indirecte (ligne 1), RELATION DIRECTE 1 et 2 définissent MOT1 et MOT2 par rapport à leur ascendant verbal commun, alors que, si MOT1 est un dérivé de MOT2 ou

vice-versa (ligne 2), RELATION DIRECTE 1 et 2 enregistrent les définitions orientée et orientée inverse appropriées. Dans le second cas, les champs RELATION INDIRECTE 1 et 2 ne sont pas renseignés. 8 318 relations de Démonette proviennent conjointement de Morphonette et de DériF. La contribution originale de Morphonette concerne 1 388 relations, les 13 180 relations restantes étant fournies par DériF, dont 8 802 relations directes (entre un nom ou un adjectif et sa base verbale). L'apport original de DériF est donc de 4 378 relations indirectes.

La ligne 3 de la table 4 illustre le type de couples que l'on trouve dans Démonette, présents dans Morphonette et absents de DériF. Le plus souvent, ces couples mettent en jeu un ascendant verbal qui n'est pas ou n'est plus attesté en synchronie (*admonition*), ou dont le sens est partiellement démotivé (*acteur* ↔ *actrice* relativement au verbe *agir*) : dans tous les cas, les champs RELATION DIRECTE 1 et 2 sont vides.

À l'inverse, et en dehors des erreurs d'implémentation de DériF, les 4 378 relations indirectes qui en sont issus et qui sont absentes de Morphonette s'expliquent par les différences entre les approches implémentées dans ces deux ressources. Les critères de filtrage de Morphonette excluent en effet les regroupements les moins fréquents comme ceux en (20) :

(20) abaissement/NOM ↔ abaisseur/NOM ; abattage/NOM ↔ abattement/NOM ;
abolissement/NOM ↔ abolitif/ADJ ; abortion/NOM ↔ avorteur/NOM

La table 5 synthétise la distribution des relations indirectes extraites, en fonction des suffixes intervenant dans la formation des couples. Nous y distinguons les noms en *-eur* formés sur le radical savant de *X*, comme *agitateur,* étiquetés 'Xteur' des autres noms en Xeur, étiquetés 'Xeur', pour lesquels la base verbale est réalisée sous la forme du radical de l'imparfait, comme *régisseur*.

Xteur	158						
Xeuse	**3 938**	134					
Xrice	338	**978**	66				
Xment	**2 226**	98	**1 390**	40			
Xage	**2 946**	80	**1 932**	11	**1 656**		
Xion	**926**	**1 544**	208	**976**	320	240	
Xif	310	448	36	304	26	24	**918**
	Xeur	Xteur	Xeuse	Xrice	Xment	Xage	Xion

TABLE 5 : Distribution des couples, en fonction des suffixes. Les effectifs supérieurs à 900 couples sont en gras.

On remarque une disparité numérique dans la réalisation de ces relations.

1. Les plus représentées reflètent une attirance particulière du couple de suffixes pour un même type de radical de base : les suffixes *-ion*, *-if* et *-rice* (dits « savants ») préfèrent les radicaux savants des verbes, alors que les suffixes *-ment* et *-age* (« non savants ») tendent à sélectionner les radicaux de l'imparfait.

2. Les couples dont l'un des membres comporte un suffixe savant et l'autre un suffixe non savant ont une fréquence qui oscille entre 200 à 900. Les couples « savants » en Xteur/Xif et Xrice/Xif se trouvent également dans cette fourchette du fait du nombre relativement faible des dérivés en *-if* dans TLFnome (1 288).

3. D'autres schémas de couples mixtes instancient peu d'entrées car ils cumulent les handicaps évoqués ci-dessus.

Ces résultats mettent en lumière les combinaisons formelles les plus utilisées pour produire les relations morphologiques entre un nom d'agent masculin et son correspondant féminin, un nom d'agent et un nom d'activité, ou encore un nom d'agent et un adjectif de propriété. Ces hypothèses, qui doivent bien entendu être validées par une projection sur un corpus authentique, fournissent un indice de la productivité potentielle des suffixations en présence, et une prédiction de la structure du lexique en devenir.

4 Évaluation

Démonette pourrait à terme servir de ressource lexicale morphologique générale pour le français. Pour estimer la pertinence de son contenu actuel, nous l'avons comparé à Verbaction, une ressource morphologique utilisée régulièrement dans différentes tâches de TAL. Le but de la comparaison est de montrer que Démonette pourrait compléter ce lexique.

Verbaction, créé l'INaLF (aujourd'hui ATILF) puis à l'ERSS (aujourd'hui CLLE), se compose de 9 386 couples verbe-nom d'action morphologiquement apparentés. Les noms sont majoritairement construits par suffixation (*consommer → consommation*), mais on y trouve également beaucoup de conversions (*collection → collectionner* ; *copier → copie*). Verbaction a été construit de manière semi-automatique. La première version (Hathout et al., 2002) a été réalisée en effectuant une analyse basée sur un apprentissage de schémas de suffixation permettant de connecter les mots potentiellement apparentés morphologiquement. Les 5 058 couples verbe-nom obtenus ont ensuite été révisés manuellement par deux lexicographes. Verbaction a ensuite été complété par 4 335 couples en utilisant la boîte à outils Webaffix (Tanguy, Hathout, 2002) pour rechercher sur le Web des couples de mots dont l'un a la forme d'un nom déverbal et l'autre celle de son verbe de base. Les couples collectés ont aussi fait l'objet d'une révision manuelle.

La comparaison de Démonette avec Verbaction n'a porté que sur leur intersection potentielle, à savoir les couples verbe-nom d'action dont les deux membres appartiennent à TLFnome et dont les noms sont suffixés en *-age*, *-ion* ou *-ment*. 4 937 couples de Démonette et 5 313 couples de Verbaction vérifient ces conditions. Nous avons ensuite réalisé un calcul simple de rappel et de précision en utilisant Verbaction comme référence. Le rappel obtenu est de 84% et la précision de 90%. 479 couples verbe-nom présents dans Démonette sont absents de Verbaction, et, réciproquement, 1 351 couples de Verbaction manquent dans Démonette.

Sur un plan qualitatif, la comparaison de Démonette avec Verbaction fait principalement ressortir trois explications pour l'absence des 479 couples dans Verbaction:

1. Dans les cas où il existe une relation de parenté morphologique entre le verbe et le nom (*ébosser → ébossage ; stagner → stagnation ; ébahir → ébahissement*), l'absence du couple dans Verbaction est un oubli qu'il faut combler.

2. Dans certains couples, le nom dérivé du verbe est concurrencé pour différentes raisons par un autre nom d'action, plus fréquent, qui appartient à un domaine de spécialité (*rattraper → rattrapement ; calibrer → calibration*) ou dénote l'un des participants à l'activité décrite par le verbe et non l'activité elle-même, par exemple, le moyen dans *embarquer → embarcation* ou *équiper → équipage*, le patient dans *vibrer → vibrion*.

3. Réciproquement, on trouve dans Démonette des couples erronés où le nom et le verbe ne sont pas apparentés (*évasion # évaser*), où le nom n'est pas déverbal (*outiller # outillage*), voire non construit en synchronie (*mentir # mention, munir # munition*), ou encore où le verbe est archaïque (*aberrer # aberration*). La comparaison avec Verbaction nous a ainsi permis de répertorier un ensemble d'analyses à corriger (essentiellement dans DériF) et par suite dans Démonette. Ces cas de bruit avéré concernent 45 des 479 couples examinés (9%).

Pour les 1 351 couples présents dans Verbaction et absents de Démonette, on observe que 281 couples constituent de véritables cas de silence comme par exemple des noms en *-ion* dont la forme est construite sur le radical savant du verbe (*adjuger → adjudication*) et des noms en *-age* dont le radical du verbe de base est formellement très éloigné de celui utilisé en flexion (*paître → pacage*). D'autres absences s'expliquent par les choix opérés lors de la constitution de Démonette ou de Verbaction. Ont été en effet exclus de Démonette les 366 mots pour lesquels DériF propose plusieurs analyses comme *préchauffage* dont la base est soit *chauffage*, soit *préchauffer*. À l'inverse, Verbaction inclut des relations de conversion nom → verbe, d'où la présence de couples comme *prédilection → prédilectionner*. Plus intéressant, le codage de Verbaction comporte des couples qui exhibent une relation de sens indubitable, mais n'entretiennent pas de relation directe base → dérivé. Par exemple, avec *chromisation*, que Verbaction apparie à *chromer* (et, plus généralement, avec les 18 couples nom → verbe dont le nom en Xisation n'a pas de base Xiser enregistrée dans le TLF) on assiste à ce qui a été appelé dans (Dal 2004) puis (Dal, Namer, à paraître) un cas de relation morphologique ternaire schématisable par <X, Xiser, Xisation> et où la parenté entre Xiser et Xisation est non orientée : la construction de *chromisation*, sur le nom *chrome*, est consubstantielle, voire antérieure à celle de *chromiser,* absent du TLF.

5 Discussion

5.1 Usages en TAL

Les applications de TAL disposent pour le français de différentes ressources lexicales munies d'informations flexionnelles comme Morphalou (Romary et al., 2004) et Lexique (New et al., 2004), syntaxiques comme Lefff (Sagot et al., 2007) ou sémantiques comme EuroWordNet (Vossen, 1998) ou Wolf (Sagot, Fišer, 2008). Certaines sont spécialisées dans le traitement de données particulières comme les noms propres dans ProLexBase (Bouchou, Maurel, 2008). L'originalité de Démonette réside dans la variété des relations morphologiques codées, et dans les annotations morpho-sémantiques qui caractérisent les unités lexicales enregistrés dans la base. Celles-ci, tout d'abord, entretiennent entre elles des relations orientées multiples : une base est associées à plusieurs dérivés, les dérivés d'une même famille sont reliés entre eux. Ensuite, chaque mot comporte, une classe sémantique et autant de définitions, donc de paraphrases, qu'il entretient de relations directes ou indirectes avec son voisinage morphologique. Enfin, cette caractéristique concerne aussi les mots simples, puisque toutes les relations sont bi-orientées.

Ces trois propriétés sont exploitables dans différents types d'application. Comme l'a signalé Clark et al. (2008), l'amélioration de l'analyse du contenu textuel passe par l'enrichissement des bases par des annotations sémantiques. De plus, Démonette facilite les tâches de sélection des mots et d'identification du sens lexical. Comme avec WordNet et Wolf, ces facultés peuvent bénéficier à toutes les applications de compréhension ou d'interprétation automatique des langues (désambiguïsation sémantique, extraction d'information), ainsi qu'aux outils destinés à la production de contenu : donner la possibilité de choisir entre plusieurs mots, identifiables par leurs bases et sélectionnés à partir de leurs étiquettes sémantiques (agent, activité, propriété), voire de choisir une paraphrase pour les remplacer, est un avantage dont peuvent tirer parti les systèmes de normalisation, de résumé, de traduction assistée ou de génération de textes. Démonette, en association avec d'autres ressources comme des bases distributionnelles (Turney, Pantel, 2010), pourra également avoir d'autres utilisations comme la désambiguïsation et la substitution lexicale (McCarthy et al., 2004), la catégorisation sémantique des textes et la recherche d'information (Tsatsaronis, Panagiotopoulou, 2009).

5.2 Montée en puissance

La version actuelle de Démonette ne couvre qu'un fragment réduit du lexique du français et n'exploite qu'une petite partie des analyses de DériF et de Morphonette. Cette couverture sera augmentée progressivement pour intégrer l'ensemble de ces analyses, mais cette extension fera apparaître des difficultés nouvelles. Il faudra notamment augmenter le nombre des types sémantiques pour permettre l'ajout de nouveaux participants (le lieu, l'instrument, etc.), de nouvelles propriétés, etc. La sélection des relations indirectes à l'intérieur des familles dérivationnelles sera également un problème difficile. En l'absence d'études connues sur ces relations indirectes, nous proposons de les limiter en nous appuyant sur des critères comme la distance dans le graphe des relations directes entre les sommets connectés pondérée par une caractérisation statistique des séquences dérivationnelles. Certaines séquences comportant deux affixes successifs sont en effet si fréquentes qu'elles sont réinterprétées par les locuteurs comme de nouveaux affixes. C'est le cas de *-aliser* (Namer, 2013 ; Hathout, Namer, 2014), de *-isation* ou de *in*X*able* (Dal, Namer, à paraître).

Un second critère corrélé au précédent consiste à considérer qu'une relation indirecte « utile » est une relation pour laquelle les locuteurs peuvent formuler facilement une définition transversale. Les relations de ce type présentes dans la version actuelle de Démonette satisfont toutes à cette condition. En revanche, d'autres relations sont plus difficiles à définir d'une façon suffisamment régulière pour être intégrer à Démonette. C'est le cas pour les couples formés d'un nom masculin et d'un nom féminin en *-et:-ette* ou en *-ier:-ière* comme *cachet:cachette* ou *boulevardier:boulevardière*. La cachette (lieu où l'on cache quelque chose) n'est pas une version féminine du cachet (sceau) et la boulevardière (prostituée) n'exerce pas la même profession que le boulevardier (auteur de pièce de théâtre de boulevard). Les critères sur la distance dans le graphe et la possibilité d'une définition transversale ne se recouvrent donc pas totalement.

D'autres couples sont également difficilement définissables l'un relativement à l'autre, comme les adverbes en *-ment* et les verbes en *-iser* construits à partir de la même base adjectivale (21), ou encore les noms Xiste formés sur une base adjectivale X et les adjectifs déverbaux correspondant Xisable (22).

(21) stérilement/R:stériliser/V ; totalement/R:totaliser/V ; verbalement/R:verbaliser/V

(22) canonisable/A:canoniste/N ; conceptualisable/A:conceptualiste/N ; individualisable/A:individualiste/N

L'intégration des mots construits par conversion et par composition sera une autre source de difficultés. Si les définitions directes des dérivations non affixales sont identiques à celles des affixations, celles des composés pourraient s'avérer plus délicates car il ne s'agit pas de relations binaires mais ternaires. Il faudra également prévoir une solution permettant de représenter les éléments de composition d'origine latine ou grecque.

Nous envisageons enfin l'intégration dans Démonette de relations morpho-sémantiques extraites de dictionnaires électroniques du domaine public en adaptant les techniques proposées par Hathout (2009).

6 Conclusion

Nous avons présenté dans cet article la première version de Démonette, une ressource lexicale comportant actuellement 31 204 relations annotées morphologiquement. Démonette est construite automatiquement à partir des résultats produits au moyen de deux approches que l'on oppose généralement l'une à l'autre en morphologie théorique : la première, Morphonette, exploite les analogies formelles entre lexèmes, et implémente les principes théoriques d'une morphologie réalisationnelle et paradigmatique. La seconde, que réalise l'analyseur DériF, est basée sur l'application de règles linguistiques orientées, conçues et validées manuellement.

Nous avons mis en pratique, avec Démonette, une conception cumulative du sens des mots dérivés envisagé comme un empilement de propriétés sémantiques partielles induites par chacune des relations dérivationnelles dans lesquelles ce mot est impliqué. Démonette est en effet un réseau où les mots sont reliés entre eux par des relations directes ou

indirectes, selon la nature de leur parenté. Les relations dérivationnelles entre deux mots sont annotées par des définitions bi-orientées permettant de construire le sens de chaque mot par rapport à celui de l'autre. Un mot est alors décrit par autant de définitions qu'il entretient de relations avec d'autres unités dans le lexique. Les mots sont par ailleurs typés sémantiquement, permettant ainsi d'estimer la contribution de chaque catégorie sémantique dans l'élaboration du réseau morphologique tissé dans Démonette. En outre, les types sémantiques sont utilisés comme paramètres pour associer aux définitions concrètes des versions abstraites qui permettent de constituer les séries dérivationnelles.

La liste des dérivations actuellement représentées dans Démontte peut facilement être étendue aux adjectifs en *-oire* comme *exploratoire* aux noms en *-oir* comme *séchoir,* aux adjectifs en *-able* comme *calculable* ou aux noms de procès ou d'état en *-ure* et *-ance/-ence* comme *gravure* et *souffrance*, tous ces dérivés étant potentiellement connectés à une base verbale.

Références

BAAYEN, R. H., PIEPENBROCK, R., GULIKERS, L. (1995). The CELEX lexical database (release 2). Distributed by the Linguistic Data Consortium, University of Pennsylvania.

BERNHARD D. (2006). Automatic Acquisition of Semantic Relationships from Morphological Relatedness. *FinTAL 2006*, Turku, Finland, Springer.

BERNHARD D., CARTONI B., TRIBOUT D. (2011). A Task-Based Evaluation of French Morphological Resources and Tools. *Linguistic Issues in Language Technology* 5(2).

BILGIN O., ÇETINOĞLU Ö ., OFLAZER K. (2004). Morphosemantic Relations In and Across Wornets. A study based on Turkish. Proceedings of GWC 2004. pp. 60-66. Brno, République Tchèque.

BOUCHOU B., MAUREL D. (2008). Prolexbase et LMF : vers un standard pour les ressources lexicales sur les noms propres. *Traitement Automatique des Langues* 49(1), 61-88.

CLARK P., FELLBAUM C., HOBBS J. R., HARRISON P., MURRAY B., THOMPSON J. (2008). Augmenting WordNet for deep understanding of text. *Proceedings of Semantics in Text Processing*, Venezia, ACL.

CREUTZ M., LAGUS K. (2005). *Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0*. Tech. Rep. A81, Helsinki University of Technology.

DAL G. (2004). *Vers une morphologie de l'évidence : d'une morphologie de l'input à une morphologie de l'output*. Villeneuve d'Ascq, Université de Lille3. Mémoire d'Habilitation à Diriger les Recherches.

DAL G., HATHOUT N., NAMER F. (1999). Construire un lexique dérivationnel: théorie et réalisations. *TALN-1999* Cargèse, Université Paris 7.

DAL G., NAMER F. (à paraître). La fréquence en morphologie : pour quels usages ?, *Langages*.

FELLBAUM C., MILLER G. A. (2003). Morphosemantic Links in WordNet. *Traitement Automatique des Langues* 44(2):69-80.

FRADIN B. (2003). *Nouvelles approches en morphologie*, Paris, PUF.

GOLDSMITH J. (2001). Unsupervised learning of the morphology of natural language. *Computational Linguistics* 27(2):153–198.

HARRIS Z. (1955). From Phoneme to Morpheme. *Language* 31(2), 190-222.

HATHOUT N. (2009). Acquisition of morphological families and derivational series from a machine readable dictionary. *Selected Proceedings of the 6th Décembrettes: Morphology in Bordeaux*. F. Montermini, G. Boyé, J. Tseng. Cambridge, Mass., Cascadilla Proceedings Project: 166-180.

HATHOUT N. (2011). Morphonette: a paradigm-based morphological network. *Lingue e linguaggio* 2011(2), 245-264.

HATHOUT, N., NAMER, F. ET DAL, G. (2002). An Experimental Constructional Database: The MorTAL Project. In Boucher, P. éditeur, *Many Morphologies*. Cascadilla, Somerville, Mass.

HATHOUT N., NAMER F. (2011). Règles et paradigmes en morphologie informatique lexématique. *TALN-2011*, Montpellier, LIRMM/ATALA.

HATHOUT N., NAMER F. (2014). Discrepancy between form and meaning in Word Formation: the case of over- and under-marking in French. *Morphology and meaning (Selected papers from the 15th International Morphology Meeting, Vienna, February 2010)* F. Rainer, W. U. Dressler, F. Gardani,H. C. Luschützky. Amsterdam, John Benjamins, 177 – 190.

KERLEROUX F. (2008). Des noms indistincts. *La raison morphologique. Hommage à la mémoire de Danielle Corbin*. B. Fradin. Amsterdam / Philadelphia, John Benjamins, 113-132.

LAFOURCADE M., JOUBERT A., 2013. Bénéfices et limites de l'acquisition lexicale dans l'expérience JeuxDeMots. In: Gala N., Zock M. (dir.), *Ressources Lexicales: Contenu, construction, utilisation, évaluation*, pp. 187-216. Linguisticae Investigationes, Supplementa 30, John Benjamins.

LEPAGE Y. (1998). Solving analogies on words: an algorithm. *17th international conference on Computational Linguistics*.

MCCARTHY D., KOELING R., WEEDS J., CARROLL. J. (2004). Finding predominant senses in untagged text. *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pp. 577-583. Barcelone.

NAMER F. (2009). *Morphologie, Lexique et TAL : l'analyseur DériF*. London, Hermes Sciences Publishing.

NAMER F. (2013). A Rule-Based Morphosemantic Analyzer for French for a Fine-Grained Semantic Annotation of Texts. *SFCM 2013*. C. Mahlow,M. Piotrowski. Heidelberg, Springer: 93-115.

NEW B., PALLIER C., BRYSBAERT M., FERRAND L. (2004). Lexique 2 : A New French Lexical Database. *Behavior Research Methods, Instruments, & Computers* 36(3), 516-524.

PALA K., HLAVÁČKOVÁ D. (2007). Derivational Relations in Czech WordNet . *Balto-Slavonic Natural Language Processing 2007* , pp 75–81, Prague . République Tchèque.

ROCHÉ M., BOYÉ G., HATHOUT N., LIGNON S., PLÉNAT M. (2011). *Des Unités Morphologiques au Lexique*. Paris, Hermès.

ROMARY L., SALMON-ALT S., FRANCOPOULO G. (2004). Standards going concrete : from LMF to Morphalou. *COLING*, Genève.

RUPPENHOFER J., BAKER C. F., FILLMORE C. J. (2002). The FrameNet Database and Software Tools. *Proceedings of the Tenth Euralex International Congress*. pp. 271-375. Copenhagen, Danemark.

SAGOT B., FIŠER S. (2008). Construction d'un wordnet libre du français à partir de ressources multilingues. *TALN-2008*, Avignon, ATALA.

STROPPA N., YVON F. (2006). Du quatrième de proportion comme principe inductif : une proposition et son application à l'apprentissage de la morphologie. *TAL* 47(1), 33-59.

TANGUY L., HATHOUT N. (2002). Webaffix :un outil d'acquisition morphologique dérivationnelle à partir du Web. *TALN-2002*, Nancy, ATALA.

TSATSARONIS, G., PANAGIOTOPOULOU, V. (2009). A generalized vector space model for text retrieval based on semantic relatedness. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pp. 70-78. Athène. Grèce.

TURNEY, P. D., PANTEL, P. (2010). From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, *37*(1), 141-188.

VOSSEN P. ed. (1998). *EuroWordNet: A Multilingual Database with Lexical Semantic Networks,*. Dordrecht, Kluwer Academic Publishers.

WATTS D. J., STROGATZ S. H. (1998). Collective dynamics of 'small-world' networks. *Nature* 393, 440-442.

Explorer le graphe de voisinage pour améliorer les thésaurus distributionnels

Vincent Claveau[1] Ewa Kijak[1] Olivier Ferret[2]
(1) IRISA - CNRS - Univ Rennes 1, Campus de Beaulieu, F-35042 Rennes
(2) CEA, LIST, Laboratoire Vision et Ingénierie des Contenus, F-91191 Gif-sur-Yvette
vincent.claveau@irisa.fr, ewa.kijak@irisa.fr, olivier.ferret@cea.fr

Résumé. Dans cet article, nous abordons le problème de construction et d'amélioration de thésaurus distributionnels. Nous montrons d'une part que les outils de recherche d'information peuvent être directement utilisés pour la construction de ces thésaurus, en offrant des performances comparables à l'état de l'art. Nous nous intéressons d'autre part plus spécifiquement à l'amélioration des thésaurus obtenus, vus comme des graphes de plus proches voisins. En tirant parti de certaines des informations de voisinage contenues dans ces graphes nous proposons plusieurs contributions.
1) Nous montrons comment améliorer globalement les listes de voisins en prenant en compte la réciprocité de la relation de voisinage, c'est-à-dire le fait qu'un mot soit un voisin proche d'un autre et vice-versa.
2) Nous proposons également une méthode permettant d'associer à chaque liste de voisins (i.e. à chaque entrées du thésaurus construit) un score de confiance.
3) Enfin, nous montrons comment utiliser ce score de confiance pour réordonner les listes de voisins les plus proches.
Ces différentes contributions sont validées expérimentalement et offrent des améliorations significatives sur l'état de l'art.

Abstract. In this paper, we address the issue of building and improving a distributional thesaurus. We first show that existing tools from the information retrieval domain can be directly used in order to build a thesaurus with state-of-the-art performance. Secondly, we focus more specifically on improving the obtained thesaurus, seen as a graph of k-nearest neighbors. By exploiting information about the neighborhood contained in this graph, we propose several contributions.
1) We show how the lists of neighbors can be globally improved by examining the reciprocity of the neighboring relation, that is, the fact that a word can be close of another and vice-versa.
2) We also propose a method to associate a confidence score to any lists of nearest neighbors (i.e. any entry of the thesaurus).
3) Last, we demonstrate how these confidence scores can be used to reorder the closest neighbors of a word.
These different contributions are validated through experiments and offer significant improvement over the state-of-the-art.

Mots-clés : thésaurus distributionnel, graphe de k proches voisins, fenêtre de Parzen, algorithme hongrois, T-normes, recherche d'information.

Keywords: distributional thesaurus, k nearest neighbor graph, Parzen window, Hungarian algorithm, T-norms, information retrieval.

1 Introduction

Les thésaurus distributionnels sont utiles à de nombreuses tâches du TAL et leur construction est un problème largement abordé depuis plusieurs années (Grefenstette, 1994). Cela reste néanmoins un champ de recherche très actif, entretenu par la mise à disposition de corpus toujours plus volumineux et de nombreuses applications. Ces thésaurus associent à chacune de leurs entrées une liste de mots qui se veulent proches sémantiquement de l'entrée. Cette notion de proximité est variable selon les travaux (synonymie, autres relations paradigmatiques, relations syntagmatiques (Budanitsky & Hirst, 2006; Adam *et al.*, 2013, pour une discussion)), mais les méthodes utilisées pour la construction automatique de ces thésaurus sont souvent partagées. Pour une grande part, ces méthodes reposent sur l'hypothèse distributionnelle de Firth (1957) : chaque mot est caractérisé par l'ensemble des contextes dans lesquels il apparaît, et la proximité sémantique de deux mots peut être déduite de la proximité de leurs contextes. Cette hypothèse a donc été mise en œuvre de différentes façons, et plusieurs pistes pour en améliorer les résultats ont été suivies (voir section suivante pour un état de l'art).

Les travaux présentés dans cet article s'inscrivent dans ce cadre, et nous proposons plusieurs contributions portant sur la création de ces thésaurus distributionnels et sur leur amélioration. Nous montrons tout d'abord que les modèles de recherche d'information (RI) sont adaptés à la tâche de création de ces thésaurus, offrant des résultats très compétitifs par rapport à l'état de l'art, tout en bénéficiant d'un outillage déjà existant (section 3).

Le cœur de notre travail se situe ensuite sur l'exploitation des relations de voisinage sémantique ainsi mesurées. Les modèles de RI fournissent en effet les listes ordonnées par similarité décroissante de tous les mots avec tous les mots, formant un graphe de plus-proches voisins. Nous proposons de tirer parti de certaines des informations de voisinage contenues dans ce graphe que nous déclinons en trois contributions.
1) Nous montrons comment améliorer globalement les listes de voisins en prenant en compte la réciprocité de la relation de voisinage, c'est-à-dire le fait qu'un mot soit un voisin proche d'un autre et vice-versa (section 4).
2) Nous proposons également une méthode permettant d'associer à chaque liste de voisins (i.e. à chaque entrée du thésaurus construit) un score de confiance (section 5). Cette méthode repose sur l'estimation, à l'aide du graphe de plus proches voisins, des probabilités de trouver un mot donné comme $i$ème voisin d'un autre.
3) Enfin, sur la base de ce travail, nous montrons comment utiliser ce score de confiance et ces probabilités pour réordonner les listes de voisins les plus proches (section 6). Pour ce faire, nous modélisons ce réordonnancement comme un problème d'optimisation de profit, résolu par l'algorithme hongrois (Kuhn & Yaw, 1955).

2 État de l'art

La notion de thésaurus distributionnel est à la fois bien connue et en même temps relativement peu abordée de façon spécifique, sans doute à cause de ses liens étroits avec la notion de similarité sémantique. Beaucoup de travaux portent sur des améliorations concernant les mesures de similarité sémantique de nature distributionnelle, c'est-à-dire directement à la construction du thésaurus. Nous les examinons dans la sous-section suivante. Mais quelques travaux présentés dans la sous-section 2.2 ont aussi cherché, une fois le thésaurus obtenu, à l'améliorer, comme nous nous proposons de le faire en l'examinant comme un graphe de plus proches voisins.

2.1 Construction des thésaurus

Si l'on considère comme point de référence le paradigme défini par Grefenstette (1994), repris à sa suite notamment par Lin (1998) et Curran & Moens (2002), une première voie d'amélioration a porté sur la pondération des éléments constitutifs des contextes distributionnels, simples mots dans le cas de cooccurrents graphiques et paires (mot, relation de dépendance syntaxique) dans le cas de cooccurrents syntaxiques. Dans cette optique, Broda *et al.* (2009) ont ainsi proposé de remplacer les poids associés aux cooccurrents par une fonction tenant compte de leur rang dans ces contextes, ce qui a l'avantage de rendre ce poids moins dépendant d'une fonction de pondération spécifique. Zhitomirsky-Geffet & Dagan (2009) opère cette modification de pondération par le biais d'un mécanisme d'amorçage en faisant l'hypothèse que les premiers voisins d'une entrée sont plus pertinents que les autres et que de ce fait, les cooccurrents qui leur sont le plus fortement associés dans leurs contextes distributionnels sont aussi plus représentatifs de l'entrée du point de vue sémantique. Le poids de ces cooccurrents est alors renforcé pour accroître leur influence lors du réordonnancement des voisins. Yamamoto & Asakura (2010) en est une variante prenant en compte un plus large ensemble de cooccurrents dans les contextes.

Au-delà des changements de pondération, certains travaux se sont attachés au contenu même des contextes distributionnels. De ce point de vue, une première distinction, opérée déjà par Grefenstette (1994) mais explorée plus en détail par Curran & Moens (2002), a été réalisée entre cooccurrents graphiques et syntaxiques, avec un avantage donné à ces derniers. Parallèlement à la nature de l'information contenue dans les contextes, la question de sa forme s'est posée en faisant l'hypothèse que l'information portée par les cooccurrents peut être représentée de façon plus dense par des dimensions sous-jacentes. Cette idée est d'ailleurs renforcée par le constat de Hagiwara *et al.* (2006), par le biais de la sélection de caractéristiques dans un cadre supervisé, que bon nombre de cooccurrents peuvent être filtrés sans altérer significativement l'identification des similarités sémantiques entre mots. Une partie des travaux visant à améliorer l'approche distributionnelle s'est donc focalisée sur l'application de méthodes de réduction de dimensions, depuis l'Analyse Sémantique Latente (Landauer & Dumais, 1997), étendue par Padó & Lapata (2007) aux cooccurrents syntaxiques, jusqu'à la factorisation de matrice non négative (Van de Cruys, 2010) en passant par le Random Indexing (Sahlgren, 2001). Ces méthodes ont cependant donné des résultats limités (Van de Cruys, 2010). Dans ce cadre, l'apprentissage de représentations distribuées

réalisées au moyen de réseaux de neurones (Huang *et al.*, 2012; Mikolov *et al.*, 2013) est également à mentionner, même si ces travaux sortent un peu du cadre distributionnel traditionnel.

2.2 Amélioration des thésaurus

Les travaux que nous considérons ici se concentrent sur des améliorations exploitant plus spécifiquement la structure du thésaurus pour en améliorer la qualité comme nous nous proposons de le faire. Zhitomirsky-Geffet & Dagan (2009) pourrait dans une certaine mesure être rattaché à cette catégorie dans la mesure où sa repondération des éléments de contexte d'un terme dépend de ses voisins sémantiques, donc de la structure du thésaurus.

Deux autres voies ont également été explorées. La première consiste à utiliser un thésaurus initial afin de sélectionner de façon non supervisée un ensemble d'exemples positifs et négatifs de termes sémantiquement similaires ou liés (Ferret, 2012, 2013b). Cet ensemble est utilisé pour entraîner un classifieur permettant ensuite de réordonner les voisins initiaux. Dans le cas de (Ferret, 2012), cette sélection est fondée sur un critère de symétrie de la relation de similarité sémantique : si A est trouvé comme voisin proche de B et B comme voisin proche de A, A et B sont probablement des exemples positifs de mots sémantiquement similaires. On retrouve là la condition de réciprocité que nous explorons dans un autre cadre en section 4. (Ferret, 2013b) s'appuie pour sa part sur l'hypothèse que des constituants similaires, au sens de leur voisinage dans un thésaurus, occupant le même rôle syntaxique dans des mots composés eux-mêmes similaires sont de probables exemples positifs de mots similaires.

La seconde approche, proposée dans (Ferret, 2013a), est plus indirecte. Elle réalise un réordonnancement des voisins sémantiques par le biais d'un processus de détection et de déclassement des voisins les moins susceptibles d'être sémantiquement liés à leur entrée. Cette détection est réalisée en appliquant un modèle discriminant de l'entrée en contexte à un échantillon des occurrences de ses voisins et en jugeant de la proximité entre entrée et voisin sur la base des décisions de ce modèle.

Comme dans ces derniers travaux, nous nous proposons d'améliorer la qualité des thésaurus produits, notamment en réordonnant les listes de voisins. Notre travail repose en partie sur des considérations proches, notamment en ce qui concerne la réciprocité, mais dans une optique différente dans laquelle les listes de plus proches voisins sont directement réordonnées en fonction des voisinages observés dans le thésaurus. En cela, nos travaux peuvent se rapprocher de ceux de (Pedronette *et al.*, 2014), faits dans un tout autre contexte applicatif (recherche d'images), mais reposant également sur l'examen des voisinages observés dans un graphe de k plus proches voisins.

3 Modèle de RI pour la construction de thésaurus distributionnels

3.1 Principes

Comme cela apparaît dans les travaux cités de l'état-de-l'art, le cœur des approches distributionnelles est de calculer des similarités entre représentations textuelles des contextes des mots étudiés. Les méthodes de calcul de similarité utilisées en recherche d'information semblent donc pertinentes pour ce problème. Pour un mot donné, l'ensemble des contextes de ses occurrences est considéré comme un document ; pour un mot w_i, on note ce document $\mathcal{C}_{w_i}$. La proximité entre deux mots est alors mesurée par une fonction de similarité RI sur leur contexte. Cette piste a beaucoup de liens avec les travaux de l'état de l'art mais semble relativement peu explorée en tant que telle, à l'exception de Vechtomova & Robertson (2012) dans le cas particulier de la recherche d'entités nommées similaires. Elle offre pourtant l'avantage d'être très facilement implémentable du fait des nombreux outils de RI disponibles.

Bien entendu, quelques adaptations doivent être faites. Dans les expériences rapportées ci-dessous, le contexte considéré est de deux mots avant et après chaque occurrence. Contrairement à la RI, on souhaite garder les mots outils, mais aussi les positions de ces mots par rapport à l'occurrence du mot examiné. Par exemple, pour l'occurrence de freedom dans l'extrait :
« ... all forms of restrictions on freedom of expression , threats ... »,
les termes d'indexation restrictions-2, on-1, of+1, expression+2 sont ajoutés à la description de freedom (i.e. sont ajoutés à $\mathcal{C}$(freedom)). Pour trouver les voisins distributionnels d'un mot, l'ensemble des contextes collectés pour ce mot sert de requête, qui est alors utilisée pour trouver les mots les plus proches (i.e. dont les contextes sont les plus proches) au sens d'une mesure de similarité RI.

Dans les expériences que nous présentons ci-dessous, nous avons testé quelques unes des mesures les plus classiquement utilisées en RI : la similarité d'Hellinger (Escoffier, 1978; Domengès & Volle, 1979), un TF-IDF/cosinus, et une similarité Okapi-BM-25 (Robertson *et al.*, 1998). Ce dernier modèle peut être vu comme une version plus moderne du TF-IDF, prenant notamment mieux en compte les différences de tailles des documents. Ce point est important puisque dans notre cas, les documents, c'est-à-dire l'ensemble des contextes d'un mot, sont effectivement de tailles très variables du fait du nombre d'occurrences lui-même très variable des différents mots. La similarité Okapi-BM25 entre un mot w_i ($\mathcal{C}(w_i)$ est vu comme une requête), et w_j ($\mathcal{C}(w_j)$ vu comme un document), s'exprime par l'équation 1 dans laquelle les composants correspondent respectivement au poids du mot considéré dans la requête, à son TF et à son IDF dans le document. qtf est le nombre d'occurrence du mot t dans le contexte de la requête $\mathcal{C}(w_i)$, et similairement tf est le nombre d'occurrences dans $\mathcal{C}(w_j)$, dl est la taille des contextes de w_j (nombre de mots dans $\mathcal{C}(w_j)$), dl_{avg} la taille moyenne des contextes, n est le nombre de documents, c'est-à-dire dans notre cas le nombre de mots examinés (nombre d'entrées du thésaurus), $df(t)$ est le nombre de contextes ($\mathcal{C}(\cdot)$) dans lesquels t apparaît, et enfin k_1, k_3 et b sont des constantes, fixées par défaut à $k_1 = 2$, $k_3 = 1000$ et $b = 0.75$.

$$\text{similarité}(w_i, w_j) = \sum_{t \in \mathcal{C}(w_i)} \frac{(k_3+1) * qtf}{k_3 + qtf} * \frac{tf * (k_1+1)}{tf + k_1 * (1 - b + b * \frac{dl(\mathcal{C}(w_j))}{dl_{avg}})} * \log \frac{n - df(t) + 0.5}{df(t) + 0.5} \quad (1)$$

Nous proposons également dans les expériences rapportées ci-dessous une version dite ajustée de la similarité Okapi-BM25, dans laquelle l'influence de la taille du document est renforcée, en prenant $b = 1$, et en mettant l'IDF au carré pour donner plus d'importance aux mots de contexte plus discriminants.

Ces modèles de RI, très classiques, ne sont pas détaillés plus avant ici ; le lecteur intéressé trouvera les notions et détails utiles dans les références citées ou des ouvrages généralistes (Boughanem & Savoy, 2008, par exemple).

3.2 Contexte expérimental

Les données et références utilisées pour nos expériences tout au long de cet article sont celles employées par Ferret (2013a) et mises à notre disposition par l'auteur. Cela nous permet d'avoir un cadre expérimental complètement comparable aux résultats publiés. Pour construire les thésaurus distributionnels, le corpus utilisé est AQUAINT-2, une collection d'articles de presse en anglais d'environ 380 millions de mots. Tous les noms de fréquence > 10 de ce corpus sont considérés, soit 25 000 noms (on note n ce nombre dans la suite) ; ils formeront les entrées des thésaurus. Le corpus est étiqueté en parties-du-discours par TreeTagger, ce qui permet de repérer les noms qui formeront les entrées du thésaurus pour nous comparer aux travaux existants. Ces informations ne sont pas utilisées pour la suite de la construction du thésaurus, ce qui assure une certaine portabilité de la méthode à d'autres langues (Freitag *et al.*, 2005).

Pour évaluer les thésaurus produits, deux références sont utilisées, soit séparément, soit conjointement. Il s'agit d'une part des synonymes de WordNet 3.0 (Miller, 1990), et d'autre part du thésaurus Moby (Ward, 1996). Ces deux ressources ont des caractéristiques assez différentes et complémentaires ; notamment, WordNet indique des liens paradigmatiques assez forts entre les mots (synonymie ou quasi-synonymie) et nous fournit ainsi des listes de 3 voisins en moyenne pour 10 473 noms du corpus, tandis que Moby regroupe des mots partageant des relations syntagmatiques ou paradigmatiques plus larges, et fournit des listes de 50 voisins en moyenne pour 9 216 noms. Les deux ressources combinées donnent une référence de 38 voisins en moyenne pour 12 243 noms. C'est cette combinaison de WordNet et Moby qui sera utilisée comme référence dans toutes les évaluations de cet article.

Cette évaluation intrinsèque porte donc sur une petite moitié des entrées du thésaurus considéré, ce que l'on peut considérer, compte tenu de sa taille, comme un ensemble d'évaluation conséquent par rapport à des jeux de test classiques (e.g. WordSim 353). Ce type d'évaluation intrinsèque est bien entendu limité par la nature des relations présentes dans les ressources de référence. Si cette limite est assez restrictive dans le cas des synonymes de WordNet, elle est beaucoup plus large et pose moins problème dans le cas de Moby, dont les relations sont très diverses.

3.3 Résultats

Pour un nom donné, notre approche par modèles de RI, comme les autres approches de l'état de l'art, ordonne les autres noms par similarité décroissante. La liste obtenue est comparée à la liste de référence, et des mesures classiques d'évaluation sont calculées. Il s'agit de la précision à divers seuils (après 1, 5, 10, 50, 100 voisins, notés P@1, P@5...), la Mean

Référence	Méthode	MAP	R-Prec	P@1	P@5	P@10	P@50	P@100
WordNet + Moby	Ferret 2013 *base*	5.6	7.7	22.5	14.1	10.8	5.3	3.8
	Ferret 2013 *best rerank*	6.1	8.4	24.8	15.4	11.7	5.7	3.8
	Hellinger	2.45	2.89	9.73	6.28	5.31	4.12	3.30
	TF-IDF	5.40	7.28	21.73	13.74	9.59	5.17	3.49
	Okapi-BM25	6.72	8.41	24.82	14.65	10.85	5.16	3.66
	Okapi-BM25 ajusté	8.97	10.94	31.05	18.44	13.76	6.46	4.54
WordNet	Ferret 2013 *base*	9.8	8.2	11.7	5.1	3.4	1.1	0.7
	Ferret 2013 *best rerank*	10.7	9.1	12.8	5.6	3.7	1.2	0.7
	Okapi-BM25 ajusté	14.17	12.22	16.97	7.10	4.47	1.41	0.84
Moby	Ferret 2013 *base*	3.2	6.7	24.1	16.4	13.0	6.6	4.8
	Ferret 2013 *best rerank*	3.5	7.2	26.5	17.9	14.0	6.9	4.8
	Okapi-BM25 ajusté	5.69	9.14	32.18	21.37	16.42	8.02	5.69

TABLE 1: Performances des modèles de RI pour la construction des thésaurus distributionnels sur la référence WordNet+Moby

Average Precision (MAP, moyenne des précisions calculées après chaque mot de la référence trouvé), la R-précision (R-prec, précision après R voisins ou R est le nombre de voisins dans la liste de référence pour le nom examiné). Elles sont toutes exprimées en pourcentage dans la suite de l'article.

Le tableau 1 recense les performances des différents modèles de similarités RI. À des fins de comparaison, nous indiquons les résultats obtenus dans les mêmes conditions par Ferret (2013a), avec d'une part une approche standard de l'état de l'art reposant sur des calculs d'information mutuelle sur les contextes, et d'autre part, une version améliorée par apprentissage (cf. section 2). Nous détaillons également certains de ces résultats selon les références WordNet et Moby prises séparément.

Plusieurs éléments méritent d'être notés dans ces premiers résultats. Notons tout d'abord que comme pour les autres travaux de la littérature, nos résultats sont globalement faibles, ce qui atteste de la difficulté de la tâche. Outre ces mesures de précision, le rappel à 100 est par exemple de 21.2 % pour la version ajusté d'Okapi sur la référence WordNet+Moby. On constate par ailleurs que certaines similarités RI sont assez peu performantes, notamment le TF seul ou la similarité d'Hellinger. Cela est peu surprenant puisque ces similarités utilisent des pondérations basiques qui ne permettent pas de mettre en valeur les contextes discriminants des mots. Les similarités incluant une notion d'IDF obtiennent en cela de meilleurs résultats. Les similarités de type Okapi-BM25 offrent de bons résultats ; la version standard d'Okapi obtient des performances similaires à l'état de l'art, et la version ajustée dépasse même largement les deux systèmes de (Ferret, 2013a), notamment en terme de qualité globale (mesurée par la MAP), quelle que soit la référence utilisée. C'est cette dernière version du système qui nous sert de référence pour la suite de cet article.

4 Réciprocité dans le graphe des k-NN

Le calcul de toutes les similarités entre toutes les paires de mots produit un graphe valué de voisinage : chaque mot est lié, avec une certaine force, aux n autres mots. Les résultats obtenus ci-dessus ne tiennent pas compte de cette structure. L'objet des sections suivantes est d'examiner comment tirer parti au mieux des relations de voisinage enfouies dans ce graphe.

Il faut préalablement noter que certaines des mesures de similarités RI que nous avons utilisées, notamment Okapi-BM25, ne sont pas symétriques. La similarité entre un mot w_i, utilisé comme requête, et un autre mot w_j ne donne pas la même valeur que la similarité entre la requête w_j et w_i. Indépendamment de cela, même pour une mesure symétrique, la relation de plus-proche voisin n'est pas non plus symétrique : un mot w_j peut-être dans les k-plus-proches voisins de w_i mais l'inverse peut être faux.

Il semble alors raisonnable de penser que la réciprocité de voisinage entre deux mots (chacun est dans les k plus proches voisins de l'autre) est tout de même un gage de confiance sur la proximité entre ces mots. L'utilisation de cette information pour améliorer les résultats précédents est examinée dans cette section. Dans la suite, on note $\tau_{w_i}(w_j)$ le rang de w_j dans la liste des voisins de w_i, qui varie donc entre 1 et n.

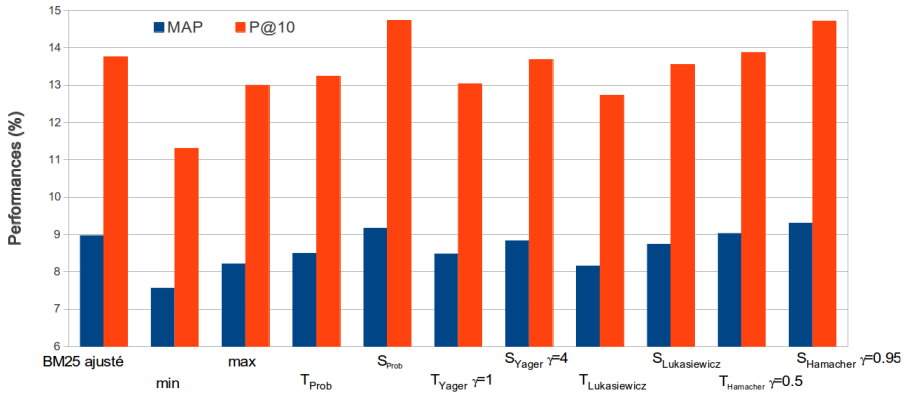


FIGURE 1: Performances de l'agrégation des rangs réciproques sur la référence WordNet+Moby

4.1 Graphe de voisinage distributionnel

La réciprocité de la relation de voisinage distributionnel a déjà été examinée et utilisée dans certains travaux (Ferret, 2013b) en sémantique distributionnelle, ou plus généralement sur des graphes de plus proches voisins (Pedronette *et al.*, 2014). Dans ces derniers travaux, la prise en compte de la réciprocité pour mener à un nouveau score de similarité a été faite simplement. Pour un mot w_i et son voisin w_j, le maximum ou le minimum des rangs ($\tau_{w_i}(w_j)$ et $\tau_{w_j}(w_i)$) est pris comme nouveau rang. Ces deux opérateurs apparaissent comme trop brutaux puisque seul l'un des rangs est pris en considération pour décider du score final, ce qui se transcrit par des performances très dégradées comme nous allons le voir.

Cependant, beaucoup d'autres opérateurs d'agrégation, avec des comportements peut-être plus appropriés à la tâche ont été proposés dans d'autres cadres, notamment en logique floue (Detyniecki, 2000, pour une revue très complète). Ces opérateurs ont une certaine sémantique permettant d'appréhender leur comportement, comme par exemple les T-normes (ET en logique floue) et S-normes (ou T-conormes, OU flou). Dans la suite de cette section, nous testons quelques uns de ces opérateurs sans prétention d'exhaustivité. Ceux-ci étant définis sur le domaine $[0, 1]^2$ et 1 représentant la certitude, ils sont utilisés pour générer un nouveau score de similarité sous la forme :

$$\text{score}_{w_i}(w_j) = \text{Agreg}(1 - \tau_{w_i}(w_j)/n, 1 - \tau_{w_j}(w_i)/n)$$

où Agreg est un opérateur d'agrégation (cf. infra pour le test de quelques fonctions possibles). Les scores obtenus sont alors utilisés pour produire une nouvelle liste de plus proches voisins de w_i (plus le score est élevé, plus la proximité sera avérée). On a bien ainsi la sémantique associée à ces opérateurs ; par exemple, si la fonction d'agrégation est $\max$, on a bien le comportement de OU flou attendu associé à cette S-norme : w_j sera classé très proche de w_i dans la nouvelle liste si w_j était proche de w_i *ou* si w_i était proche de w_j. Pour la T-norme $\min$, il faut que w_j soit proche de w_i *et* que w_i soit proche de w_j.

4.2 Résultats

Pour la fonction d'agrégation Agreg, outre le $\min$ et le $\max$, nous rapportons dans la figure 1 les résultats obtenus avec les T-normes (ou familles de T-normes dépendant d'un paramètre γ) suivantes :

$\text{T}_{\text{Prob}}(x, y) = x * y$

$\text{T}_{\text{Lukasiewicz}}(x, y) = \max(x + y - 1, 0)$

$\text{T}_{\text{Hamacher}}(x, y) = \frac{x*y}{\gamma+(1-\gamma)*(x+y-x*y)}$ with $\gamma \geq 0$

$\text{T}_{\text{Yager}}(x, y) = \max(0, 1 - \sqrt[\gamma]{(1-x)^\gamma + (1-y)^\gamma})$ with $\gamma > 0$

Nous testons aussi les S-normes associées, obtenues par généralisation de la loi de De Morgan : $S(x, y) = 1-T(1-x, 1-y)$. Pour les familles de T-normes dépendant d'un paramètre, nous avons fait varier ce dernier de manière systématique et les résultats rapportés sont ceux maximisant la MAP.

On remarque que ces opérateurs obtiennent des résultats très divers. Ceux qui induisent un seuil (i.e. pour certaines valeurs de $\tau_{w_i}(w_j)$ et $\tau_{w_j}(w_i)$, ces opérateurs renvoient une valeur par défaut générant trop d'ex æquo parmi les voisins, comme le min, max, les normes Lukasiewicz, et d'autres pour certains γ) dégradent la qualité des listes de plus proches voisins.

Référence	MAP	R-Prec	P@1	P@5	P@10	P@50	P@100
WordNet + Moby	9.30 (+3.75)	11.06 (+2.03)	30.42 (-2.53)	19.29 (+4.58)	14.71 (+6.92)	7.09 (+9.78)	4.86 (+7.07)
WordNet	15.05 (+6.23)	12.81 (+4.81)	17.55 (+3.41)	7.96 (+12.16)	5.07 (+13.30)	1.63 (+15.69)	0.94 (+12.23)
Moby	5.90 (+3.65)	11.86 (+4.14)	*31.77 (-1.27)*	21.65 (+1.34)	17.0 (+3.53)	8.42 (+5.01)	5.92 (+4.12)

TABLE 2: Performances et gains (%) par agrégation des rangs réciproques sur les références WordNet et Moby séparément avec une agrégation des rangs par S_{Hamacher} $\gamma = 0.95$

Les T-normes, privilégiant les paires de mots proches l'une de l'autre dans les deux sens, sont trop contraignantes. Cela rejoint les conclusions des travaux cités : la condition de réciprocité, appliquée trop strictement, ne permet pas d'améliorer les listes de plus proches voisins sur l'ensemble des mots. En revanche, les S-normes semblent mieux à même de tirer parti du classement. Les améliorations sont dans ce cas modestes en terme de qualité globale (MAP), mais importantes à certains rangs (P@10, P@50).

Enfin, il est important de noter que ces résultats dépendent beaucoup de la ressource utilisée comme référence, comme nous l'illustrons dans le tableau 2 en testant l'agrégation avec S_{Hamacher} $\gamma = 0.95$ sur Moby et WordNet séparément. Pour s'assurer que les différences soient statistiquement significatives, nous effectuons un test de Wilcoxon ($p < 0.05$) (Hull, 1993) ; les résultats non significatifs sont indiqués en italiques. Sur WordNet, basée sur une relation de synonymie assez forte et donc réciproque, les gains obtenus par notre approche sont bien plus importants que sur la référence issue de Moby.

5 Estimation de la confiance d'une liste de voisins distributionnels

Dans la section précédente, le rang de w_i dans la liste des voisins de w_j est utilisé pour améliorer le classement de w_j dans la liste des voisins de w_i. On peut aussi s'intéresser d'une manière plus générale aux positions relatives de w_i et w_j dans toutes les listes de voisins de tous les mots pour en tirer une information peut-être plus complète. Dans un premier temps, nous proposons d'en tirer un critère de confiance associé à chaque liste de plus-proches voisins en se basant uniquement sur des éléments du graphe de voisinage.

5.1 Principe

On fait l'hypothèse suivante : la liste de plus proches voisins d'un mot w est probablement de bonne qualité si la proximité (en terme de rang) entre w et chacun de ses voisins w_i est cohérente avec la proximité observée entre ces mêmes mots (w, w_i) dans les listes de voisins d'autres mots. L'intuition est que des mots supposés proches doivent aussi se retrouver proches des mêmes mots. Par exemple, si w_i est un voisin très proche de w, et que w est un voisin très proche de w_j, on s'attend à ce que w_i soit aussi très proche de w_j. Si les k plus proches voisins de w ont cette qualité, alors on accorde une certaine confiance à cette liste de voisins.

Formellement, nous définissons en terme probabiliste l'indice de confiance de la liste des k plus proches voisins de w par :

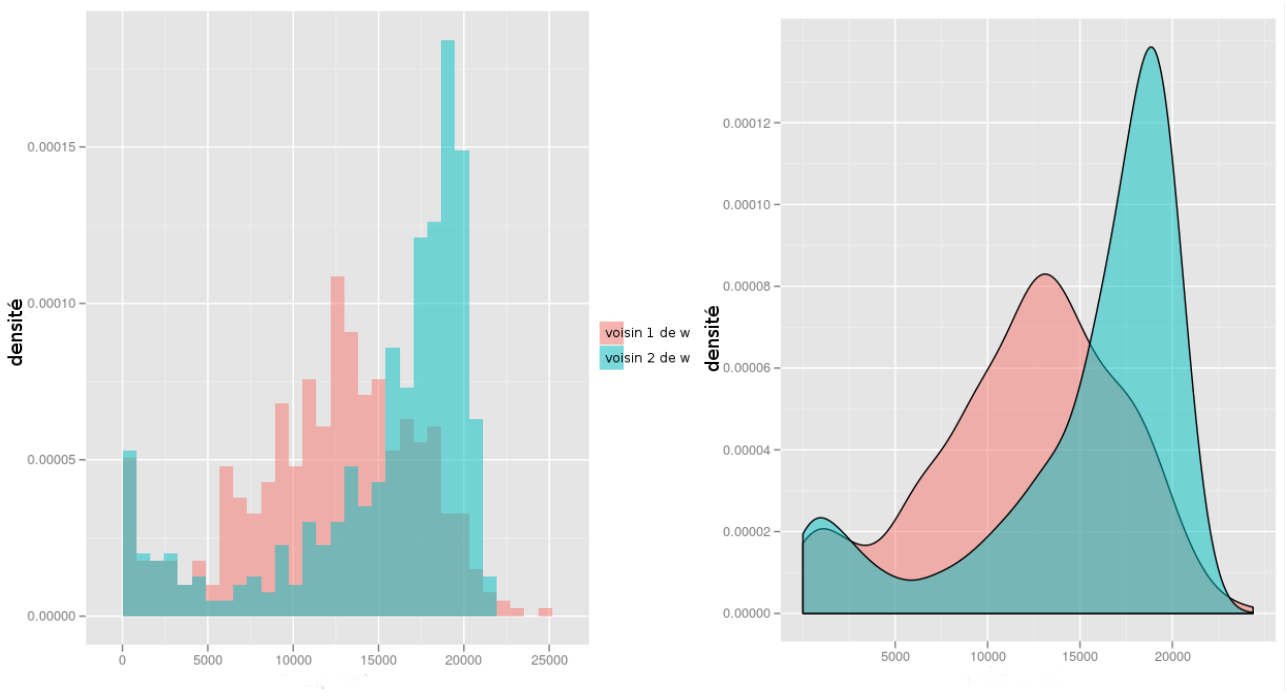
$$Q(w) = \prod_{\{w_i | \tau_w(w_i) \leq k\}} p(\delta(w, w_i) = \tau_w(w_i))$$

avec $p(\delta(w, w_i) = \tau_w(w_i))$ la probabilité que w_i soit le $\tau_w(w_i)$ème voisin de w (i.e. l'écart en terme de nombre de voisins, noté $\delta(w, w_i)$, est de $\tau_w(w_i)$).

Le problème est alors d'estimer pour chaque couple de mots (w, w_i) la distribution de probabilité $p(\delta(w, w_i))$. On utilise pour cela une méthode d'estimation de densité non-paramétrique par fenêtre de Parzen. Nous décrivons comment cette méthode classique (Parzen, 1962; Wasserman, 2005) est appliquée dans notre cas ci-après.

5.2 Estimation par fenêtres de Parzen

Soit x_{ab} la distance (différence entre les rangs) entre deux mots w_a et w_b dans une liste des voisins d'un mot quelconque. On dispose d'un échantillon de n réalisations de x_{ab}, supposées *iid* : (x^1_{ab}, x^2_{ab}, ... , x^n_{ab}), qui sont les distances observées



(a) (b)

FIGURE 2: (a) Deux histogrammes (en bleu et en rouge) des écarts d'un mot et deux de ses voisins. (b) Deux densités de probabilités (en bleu et en rouge) estimées par fenêtres de Parzen, correspondant aux données de la figure 2a.

entre w_a et w_b dans chacune des listes (complètes) des voisins de chaque mot. On les suppose *iid* ; cette hypothèse n'est certainement pas remplie puisque ces distances sont calculées en partie sur des contextes similaires, mais cette simplification permet une formalisation simple et efficace.

En effet, on peut alors estimer la densité de probabilité de x_{ab} avec la technique des fenêtres de Parzen grâce à un estimateur à noyau (équation 2) avec h un paramètre de lissage à fixer, et K un noyau permettant d'estimer la densité localement.

$$\widehat{p_h}(x_{ab}) = \frac{1}{n\,h}\sum_{i=1}^{n} K\left(\frac{x_{ab}-x_{ab}^i}{h}\right) \quad (2) \qquad \text{avec} \quad K(u) = \frac{1}{\sqrt{2\pi}} e^{\frac{-u^2}{2}} \quad (3) \qquad \text{et} \quad \hat{h} = 0.9\ \min(\hat{\sigma}, \frac{q_3-q_1}{1.34})\ n^{-\frac{1}{5}} \quad (4)$$

Dans notre cas, nous choisissons classiquement un noyau gaussien (équation 3). La probabilité résultante est donc un mélange de gaussiennes centrées réduites sur chaque x^i. Il est montré que le choix du noyau a une influence réduite sur l'estimation. En revanche, ces méthodes sont connues pour être sensibles au choix du paramètre de lissage h, qui contrôle la régularité de l'estimation. Son choix crucial est un problème particulièrement difficile, mais largement abordé dans la littérature. Pour le fixer, nous utilisons la règle empirique de Silverman (Silverman, 1986, page 48, eqn (3.31)). Sous l'hypothèse de normalité de la distribution sous-jacente, cette règle propose une façon simple de calculer le paramètre h optimum lorsque des fonctions gaussiennes sont utilisées pour approximer des données univariées (équation 4 où $\hat{\sigma}$ est l'écart type estimé sur l'échantillon, q_1 et q_3 respectivement les premier et troisième quartiles).

Une fois ces probabilités estimées sur chacun des k plus proches voisins de w, on peut alors calculer le score de confiance $Q(w)$. La complexité de ce calcul pour l'ensemble des listes de voisinage est donc en $\mathcal{O}(k * n^2)$.

5.3 Utilité du score de confiance

L'intérêt attendu du score de confiance est de permettre d'avoir un indice a priori de la qualité d'une liste de voisins pour un mot donné. Un tel score peut ainsi être utile pour de nombreuses applications exploitant les thésaurus produits par notre approche. Une évaluation du score de confiance par le biais de telles applications serait certainement le plus adapté,

mais dépasse le cadre de cet article. Nous utilisons à défaut une évaluation directe vis-à-vis de la MAP : nous mesurons la corrélation entre la MAP et le score de confiance, l'idée étant qu'une entrée avec une liste de voisins de faible qualité correspond à une entrée ayant une MAP faible.

Plusieurs indices de corrélation peuvent être employés. L'indice r de Pearson mesure une corrélation linéaire entre score et MAP. Nous utilisons également les corrélations ρ de Spearman et τ de Kendall qui ne font pas d'hypothèse de linéarité et comparent uniquement l'ordre des mots classés selon la MAP à l'ordre selon le score de confiance. Les résultats de ces trois coefficients sont donnés dans le tableau 3 (1 indique une corrélation parfaite, 0 une absence de corrélation et -1 une corrélation inverse), avec pour chacun la p-valeur du test de significativité associé (une p-valeur faible, par exemple < 0.05, indique un résultat statistiquement significatif). Les scores de confiance sont obtenus avec $k = 20$; d'autres expériences non rapportées ici montrent que ce paramètre, s'il est choisi entre 5 et 100, influence peu les valeurs de corrélation. Ces mesures montrent une corrélation certaine et statistiquement significative entre notre score de confiance et la MAP, mais néanmoins imparfaite et non linéaire. Le score de confiance est tout de même un bon indicateur de qualité comme en témoigne aussi le graphe en figure 3 où est représentée la moyenne des MAP (en ordonné) sur les listes de voisins ayant un score de confiance inférieur à un seuil que nous faisons varier (en abscisse).

Coefficient de corrélation	valeur	significativité statistique
Pearson r	0.16	$p < 10^{-40}$
Kendall τ	0.37	$p < 10^{-64}$
Spearman ρ	0.51	$p < 10^{-64}$

TABLE 3: Mesures de corrélation entre le coefficient de confiance et la MAP, avec leur significativité (p-valeur)

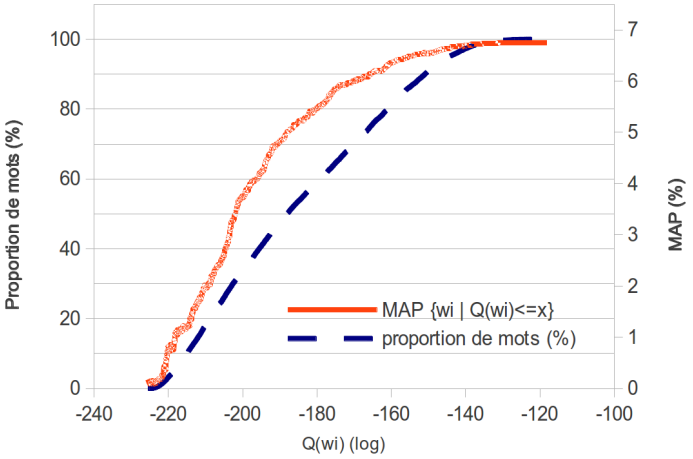


FIGURE 3: MAP des mots dont le score de confiance est inférieur à un certain seuil (donné en abscisse (log)) et proportion cumulée de mots concernés

Le score de confiance peut être utilisé pour améliorer les résultats des techniques d'agrégation vus en section 4. L'idée est simplement d'intégrer le score de confiance dans le score final :
$\text{score}_{w_i}(w_j) = Q(w_j) * \text{Agreg}(1 - \tau_{w_i}(w_j)/n, 1 - \tau_{w_j}(w_i)/n)$
Comme on le voit dans le tableau 4, l'ajout de cette information permet des gains encore plus importants que ceux rapportés dans la section précédente. Comme précédemment, ces gains sont plus sensibles en fin de liste (P@50, P@100). Dans la section suivante, nous tentons d'améliorer également les résultats en début de liste, c'est-à-dire sur les voisins jugés les plus proches, en utilisant différemment les scores de confiance.

Méthode	MAP	R-Prec	P@1	P@5	P@10	P@50	P@100
$S_{\text{Hamacher}}\ \gamma = 0.95$	9.61 (+7.20)	11.59 (+5.85)	*30.86 (-0.53)*	19.52 (+5.83)	14.76 (+7.24)	7.03 (+8.88)	4.93 (+8.67)

TABLE 4: Performances et gains (%) par agrégation des rangs réciproques prenant en compte le score de confiance sur la référence WordNet+Moby

6 Réordonnancement local

La méthode précédente donne un score global à la liste, mais on peut aussi exploiter les probabilités de classements individuelles (les $p(\delta(w_i, w_j))$) calculées selon la méthode des fenêtres de Parzen. Pour un mot donné w, on dispose pour chacun de ses voisins w_j d'un score de confiance individuel lié à son rang actuel ($p(\delta(w, w_j)) = \tau_w(w_j)$), et l'on peut

également calculer les probabilités de voir ce voisin à n'importe quel autre rang τ (probabilité que ce mot soit au rang 1, 2...). Dans cette section, on se propose d'utiliser ces informations plus locales pour améliorer les résultats en réordonnant les k-plus-proches voisins.

6.1 Réordonner par l'algorithme hongrois

Une première approche consisterait à réordonner la liste sur la base de ce critère, des voisins les plus probables aux moins probables. Mais notre critère de qualité associé à chaque mot est imparfait, et un tel réordonnancement dégrade fortement les résultats. On propose donc à la place une méthode permettant de réordonner les k-plus-proches voisins de manière locale (un mot qui n'était pas dans les k-plus-proches ne peut pas y entrer) et contrôlée (un mot ne peut pas s'éloigner trop de son rang initial).

Notre problème s'exprime par la matrice suivante, dite matrice de profit, dans laquelle les lignes correspondent aux mots dans l'ordre du classement actuel (notés w_1 à w_k), et les colonnes correspondent aux nouveaux rangs auxquels assigner ces mots. Étant données les probabilités de chaque mot w_j d'apparaître à un rang τ, l'objectif est de trouver la permutation des k plus proches voisins la plus probable, c'est-à-dire celle qui "profite" le plus.

$$\mathcal{M}_{\text{profit}} = \begin{pmatrix} p(\delta(w,w_1)=1) & \cdots & p(\delta(w,w_1)=k) \\ \vdots & \ddots & \vdots \\ p(\delta(w,w_k)=1) & \cdots & p(\delta(w,w_k)=k) \end{pmatrix} \qquad \mathcal{M}_{\text{pénalité}} = \begin{pmatrix} 1 & \frac{k-1}{k} & \cdots & 0 \\ \frac{k-1}{k} & 1 & \cdots & \frac{1}{k} \\ \vdots & \ddots & \vdots & \\ 0 & \frac{1}{k} & \cdots & 1 \end{pmatrix}$$

Comme nous l'avons souligné, on souhaite par ailleurs limiter les déplacements importants, pour éviter qu'un voisin initialement très proche se retrouve beaucoup plus loin et inversement. On ajoute cette contrainte à la matrice de profit en prenant le produit d'Hadamard (produit de matrices composante par composante, noté $\circ$) avec la matrice de pénalité $\mathcal{M}_{\text{pénalité}}$.

On se trouve alors face à un problème d'optimisation combinatoire, qui peut se résoudre en un temps polynomial en appliquant l'algorithme hongrois (Kuhn & Yaw, 1955, pour une description de l'algorithme) sur $\mathcal{M}_{\text{profit}} \circ \mathcal{M}_{\text{pénalité}}$. Cet algorithme a été initialement proposé pour optimiser l'assignation de travailleurs (dans notre cas les voisins) sur des tâches (dans notre cas, le rang), selon le profit dégagé par chaque travailleur pour chaque tâche (ici, la probabilité que ce voisin soit à ce rang). Il permet de trouver l'assignation optimale étant donnée une matrice de profit. Son résultat nous indique donc un nouveau rang pour chaque mot. L'algorithme converge sur une solution optimale et est de complexité $\mathcal{O}(k^3)$ (pour le réordonnancement des k-plus-proches voisins).

6.2 Résultats

Le tableau 5 présente les performances obtenues par rapport à notre référence Okapi-BM25 ajusté selon les mêmes modalités expérimentales que précédemment. Comme précédemment, on a fixé le voisinage considéré à $k = 20$; les précisions au delà de ce seuil sont donc inchangées et ne sont pas reportées. Nous testons l'efficacité de ce réordonnancement sur l'ensemble des listes de voisins et sur le tiers des listes ayant les scores de qualité les plus faibles. Il en ressort que le réordonnancement sur l'ensemble des listes n'apporte pas de véritable gain ; en revanche, sur les listes dont le score de confiance est faible, le gain est substantiel. Par ailleurs, contrairement aux expériences de la section 4, ces gains portent par construction sur les têtes de listes, qui sont les plus à même d'être utilisées en pratique. La différence entre le traitement de l'ensemble des mots et celui sur le tiers ayant le score de confiance le plus bas s'explique de deux manières. D'une part, les listes ayant les plus forts scores de confiance correspondent en majeure partie aux listes ayant les meilleures MAP, comme attendu (et illustré en figure 3). Celles-ci laissent donc a priori peu de marge à l'amélioration. D'autre part, indé-

Méthode	MAP	R-Prec	P@1	P@5	P@10
tous les mots	9.16 (+2.17)	11.24 (+2.76)	*30.73 (-1.02)*	19.30 (+4.64)	14.37 (+4.44)
tiers avec le plus faible $Q(w_i)$	9.55 (+6.44)	11.81 (+7.99)	31.85 (+2.56)	20.43 (+10.81)	15.46 (+12.37)

TABLE 5: Performances et gains (%) du réordonnancement par l'algorithme hongrois

pendamment de leur MAP, on peut aussi supposer que ces mêmes listes ont déjà un arrangement optimal des probabilités individuelles qui explique le fort score de confiance, le réordonnancement ne concernant alors que peu de voisins.

7 Conclusion et perspectives

Les différentes contributions proposées dans cet article ne se placent pas toutes au même niveau. La construction de thésaurus en utilisant des outils issus de la RI n'est pas une innovation conceptuelle majeure, mais cette approche semble curieusement inexplorée bien qu'elle fournisse des résultats très compétitifs en demandant un minimum de travail de mise-en-œuvre grâce aux outils existants de la RI.

Les différentes propositions exploitant le graphe de voisinage pour améliorer le thésaurus relèvent d'une démarche plus originale où l'ensemble du thésaurus est considéré. Nous y avons en particulier examiné les aspects de réciprocité et de distance, en terme de rang, entre deux mots pour proposer plusieurs contributions. Certaines hypothèses, comme la réciprocité, se défendent aisément pour des relations comme la synonymie, mais restent à valider pour des relations plus complexes. À ce titre, une analyse plus fine par type de relations en s'appuyant sur la typologie de Moby reste à faire. Cependant, les améliorations apportées par l'agrégation sur l'ensemble des voisins ou la technique de réordonnancement à partir des scores de confiance valident globalement notre démarche. Il convient de noter à ce propos que les gains obtenus sont petits en valeur absolue, mais constituent, par rapport à ceux observés dans le domaine, des améliorations significatives. Une analyse contrastive entre Moby et WordNet apporterait également des éléments intéressants et complémentaires à Ferret (2013b).

Les différents aspects de ce travail ouvrent de nombreuses pistes de recherche. Par exemple, beaucoup d'autres fonctions d'agrégation outre celles testées en section 4 existent dans la littérature. Certaines pourraient d'ailleurs offrir la possibilité d'incorporer le score de confiance associé à chaque voisin, comme les intégrales de Choquet ou de Sugeno (Detyniecki, 2000). Plus largement, il serait intéressant d'utiliser itérativement les améliorations des listes de voisins pour mettre à jour les scores de confiance, etc., en s'inspirant par exemple de ce qui est proposé par Pedronette *et al.* (2014). Au delà des thésaurus distributionnels, les méthodes proposées pour calculer des scores de confiance ou réordonner les listes de voisins peuvent s'appliquer à d'autres problèmes où ces graphes de k plus proches voisins sont construits. Notons également que nous n'avons considéré qu'une petite partie de l'information portée par le graphe de voisinage. Nous nous sommes concentrés sur les aspects de réciprocité, mais d'autres travaux, prenant en compte d'autres aspects de ce graphe (la transitivité notamment, ou plus globalement sa topologie), pourraient mener à d'autres améliorations.

Références

ADAM C., FABRE C. & MULLER P. (2013). Évaluer et améliorer une ressource distributionnelle : protocole d'annotation de liens sémantiques en contexte. *TAL*, **54**(1), 71–97.

M. BOUGHANEM & J. SAVOY, Eds. (2008). *Recherche d'information : états des lieux et perspectives*. http ://www.editions-hermes.fr/ : Hermès Science.

BRODA B., PIASECKI M. & SZPAKOWICZ S. (2009). Rank-Based Transformation in Measuring Semantic Relatedness. In *22nd Canadian Conference on Artificial Intelligence*, p. 187–190.

BUDANITSKY A. & HIRST G. (2006). Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, **32**(1), 13–47.

CURRAN J. R. & MOENS M. (2002). Improvements in automatic thesaurus extraction. In *Workshop of the ACL Special Interest Group on the Lexicon (SIGLEX)*, p. 59–66, Philadelphia, USA.

DETYNIECKI M. (2000). *Mathematical aggregation operators and their application to video querying*. PhD thesis, Université de Paris 6.

DOMENGÈS D. & VOLLE M. (1979). Analyse factorielle sphérique : une exploration. *Annales de l'INSEE*, **35**, 3–83.

ESCOFFIER B. (1978). Analyse factorielle et distances répondant au principe d'équivalence distributionnelle. *Revue de statistique appliquée*, **26**(4), 29–37.

FERRET O. (2012). Combining bootstrapping and feature selection for improving a distributional thesaurus. In *20th European Conference on Artificial Intelligence (ECAI 2012)*, p. 336–341, Montpellier, France.

FERRET O. (2013a). Identifying bad semantic neighbors for improving distributional thesauri. In *51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, p. 561–571, Sofia, Bulgaria.

FERRET O. (2013b). Sélection non supervisée de relations sémantiques pour améliorer un thésaurus distributionnel. In *20ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2013)*, p. 48–61, Les Sables d'Olonne, France.

FIRTH J. R. (1957). *Studies in Linguistic Analysis*, chapter A synopsis of linguistic theory 1930-1955, p. 1–32. Blackwell : Oxford.

FREITAG D., BLUME M., BYRNES J., CHOW E., KAPADIA S., ROHWER R. & WANG Z. (2005). New experiments in distributional representations of synonymy. In *Ninth Conference on Computational Natural Language Learning (CoNLL)*, p. 25–32, Ann Arbor, Michigan, USA.

GREFENSTETTE G. (1994). *Explorations in automatic thesaurus discovery*. Kluwer Academic Publishers.

HAGIWARA M., OGAWA Y. & TOYAMA K. (2006). Selection of effective contextual information for automatic synonym acquisition. In *21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006)*, p. 353–360, Sydney, Australia.

HUANG E. H., SOCHER R., MANNING C. D. & NG A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *50th Annual Meeting of the Association for Computational Linguistics (ACL'12)*, p. 873–882.

HULL D. (1993). Using Statistical Testing in the Evaluation of Retrieval Experiments. In *Proc. of the 16th Annual ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'93*, Pittsburgh, États-Unis.

KUHN H. W. & YAW B. (1955). The hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, **2**, 83–97.

LANDAUER T. K. & DUMAIS S. T. (1997). A solution to Plato's problem : the latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, **104**(2), 211–240.

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

MIKOLOV T., YIH W.-T. & ZWEIG G. (2013). Linguistic regularities in continuous space word representations. In *2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL HLT 2013)*, p. 746–751, Atlanta, Georgia.

MILLER G. A. (1990). WordNet : An On-Line Lexical Database. *International Journal of Lexicography*, **3**(4).

PADÓ S. & LAPATA M. (2007). Dependency-based construction of semantic space models. *Computational Linguistics*, **33**(2), 161–199.

PARZEN E. (1962). On estimation of a probability density function and mode. *Ann. Math. Stat.*, **33**, 1065–1076.

PEDRONETTE D. C. G., PENATTI O. A. & DA S. TORRES R. (2014). Unsupervised manifold learning using reciprocal knn graphs in image re-ranking and rank aggregation tasks. *Image and Vision Computing*, **32**(2), 120 – 130.

ROBERTSON S. E., WALKER S. & HANCOCK-BEAULIEU M. (1998). Okapi at TREC-7 : Automatic Ad Hoc, Filtering, VLC and Interactive. In *Proc. of the 7th Text Retrieval Conference, TREC-7*, p. 199–210.

SAHLGREN M. (2001). Vector-based semantic analysis : Representing word meanings based on random labels. In *ESSLLI 2001 Workshop on Semantic Knowledge Acquisition and Categorisation*, Helsinki, Finland.

SILVERMAN B. W. (1986). *Density estimation for statistics and data analysis*. Monographs on statistics and applied probability. London, Glasgow, Weinheim : Chapman and Hall Boca Raton.

VAN DE CRUYS T. (2010). *Mining for Meaning. The Extraction of Lexico-semantic Knowledge from Text*. PhD thesis, University of Groningen, The Netherlands.

VECHTOMOVA O. & ROBERTSON S. (2012). A domain-independent approach to finding related entities. *Information Processing and Management*, **48**(4).

WARD G. (1996). Moby thesaurus. Moby Project.

WASSERMAN L. (2005). *All of Statistics : A Concise Course in Statistical Inference*. Springer Texts in Statistics.

YAMAMOTO K. & ASAKURA T. (2010). Even unassociated features can improve lexical distributional similarity. In *Second Workshop on NLP Challenges in the Information Explosion Era (NLPIX 2010)*, p. 32–39, Beijing, China.

ZHITOMIRSKY-GEFFET M. & DAGAN I. (2009). Bootstrapping Distributional Feature Vector Quality. *Computational Linguistics*, **35**(3), 435–461.

Réduction de la dispersion des données par généralisation des contextes distributionnels : application aux textes de spécialité

Amandine Périnet[1] Thierry Hamon[2,3]
(1) INSERM, U1142, LIMICS, F-75006, Paris, France;
Sorbonne Universités, UPMC Univ Paris 06, UMR_S 1142, LIMICS, F-75006, Paris, France;
Université Paris 13, Sorbonne Paris Cité, LIMICS, (UMR_S 1142), F-93430, Villetaneuse, France.
amandine.perinet@edu.univ-paris13.fr
(2) LIMSI-CNRS, 91403 Orsay, France
(3) Université Paris 13, Sorbonne Paris Cité, 93430 Villetaneuse, France
hamon@limsi.fr

Résumé. Les modèles d'espace vectoriels mettant en œuvre l'analyse distributionnelle s'appuient sur la redondance d'informations se trouvant dans le contexte des mots à associer. Cependant, ces modèles souffrent du nombre de dimensions considérable et de la dispersion des données dans la matrice des vecteurs de contexte. Il s'agit d'un enjeu majeur sur les corpus de spécialité pour lesquels la taille est beaucoup plus petite et les informations contextuelles moins redondantes. Nous nous intéressons au problème de la limitation de la dispersion des données sur des corpus de spécialité et proposons une méthode permettant de densifier la matrice en généralisant les contextes distributionnels. L'évaluation de la méthode sur un corpus médical en français montre qu'avec une petite fenêtre graphique et l'indice de Jaccard, la généralisation des contextes avec des relations fournies par des patrons lexico-syntaxiques permet d'améliorer les résultats, alors qu'avec une large fenêtre et le cosinus, il est préférable de généraliser avec des relations obtenues par inclusion lexicale.

Abstract. Vector space models implement the distributional hypothesis relying on the repetition of information occurring in the contexts of words to associate. However, these models suffer from a high number of dimensions and data sparseness in the matrix of contextual vectors. This is a major issue with specialized corpora that are of much smaller size and with much lower context frequencies. We tackle the problem of data sparseness on specialized texts and we propose a method that allows to make the matrix denser, by generalizing of distributional contexts. The evaluation of the method is performed on a French medical corpus, and shows that with a small graphical window and the Jaccard Index, the context generalization with lexico-syntactic patterns improves the results, while with a large window and the cosine measure, it is better to generalize with lexical inclusion.

Mots-clés : Analyse distributionnelle, textes de spécialité, hyperonymie, dispersion des données, modèle d'espace vectoriel, méthode hybride.

Keywords: Distributional analysis, specialized texts, hypernymy, data sparseness, Vector Space Model, hybrid method.

1 Introduction

L'analyse distributionnelle (AD) s'appuie sur l'hypothèse que des mots apparaissant dans des contextes similaires ont tendance à être proches sémantiquement (Harris, 1954; Firth, 1957). Cette hypothèse est généralement mise en œuvre grâce à des modèles d'espace vectoriel (VSM) où les vecteurs représentent à la fois ces informations contextuelles mais également des données statistiques distributionnelles (Sahlgren, 2006). Chaque mot cible d'un texte est représenté comme un point dans un espace mathématique en fonction de ses propriétés distributionnelles dans le texte (Turney & Pantel, 2010; Lund & Burgess, 1996). La similarité sémantique entre deux mots est alors définie comme une proximité dans un espace à n-dimensions où chaque dimension correspond à des contextes partagés possibles. Les VSM ont ainsi l'avantage de permettre une quantification facile de la proximité sémantique entre deux mots en mesurant la distance entre deux vecteurs au sein de cet espace (par ex. le cosinus de leur angle). Cependant, outre le nombre important de dimensions (à titre d'exemple, Sahlgren (2006) manipule des VSM allant jusqu'à plusieurs millions de dimensions), les VSM souffrent également de la dispersion des données dans la matrice représentant l'espace vectoriel (Chatterjee & Mohan, 2008) :

beaucoup d'éléments de la matrice sont à 0 car peu de contextes sont associés à un mot cible. Cet inconvénient est dû notamment à la distribution des mots dans le corpus (Baroni *et al.*, 2009) : quelle que soit la taille du corpus, la plupart des mots ont des fréquences basses et un nombre de contextes très limité au regard du nombre de mots dans le corpus. Ces deux derniers points rendent difficile le calcul de la similarité entre deux mots. En conséquence, les méthodes basées sur l'analyse distributionnelle obtiennent de meilleures performances lorsque beaucoup d'informations sont disponibles, et notamment sur des corpus de langue générale, en général très volumineux (Weeds & Weir, 2005; van der Plas, 2008). Mais, la réduction de la dispersion des données reste un aspect important sur des corpus de langue générale. Elle est aussi un problème majeur lorsque l'on travaille sur des corpus de spécialité. En effet, ces corpus se caractérisent par des tailles beaucoup plus petites, avec des fréquences et un nombre de contextes différents d'autant plus faibles. Nous nous intéressons à ce dernier point en proposant l'adaptation d'une méthode d'analyse distributionnelle qui permette d'obtenir de meilleurs résultats sur des textes de spécialité. Pour cela, nous avons cherché à réduire la diversité des contextes en les généralisant. Il est alors possible d'augmenter la fréquence des contextes distributionnels qui résultent de cette généralisation et ainsi réduire la dispersion des données et la dimension de l'espace vectoriel. Nous présentons ici une méthode à base de règles permettant la généralisation des contextes distributionnels à l'aide de résultats issus de méthodes d'acquisition de relations sémantiques. Nous adaptons les paramètres de la méthode distributionnelle utilisée aux corpus de spécialité, en intégrant notamment ces contextes généralisés.

Dans la suite de l'article, nous présentons tout d'abord un état de l'art des méthodes de réduction de la dispersion des données dans les méthodes distributionnelles (section 2). Nous décrivons ensuite le corpus et les relations utilisés (section 3), la méthode de généralisation de contextes proposée (section 4), ainsi que les expériences réalisées pour évaluer son impact sur un corpus de spécialité (section 5). Les résultats sont évalués puis analysés en terme de précision, de R-précision et de MAP (section 6).

2 État de l'art

La réduction de la dispersion des données est un enjeu majeur en analyse distributionnelle. Pour cela, les méthodes proposées visent à influer sur la sélection des contextes utiles ou à intégrer des informations sémantiques de manière à modifier la distribution des contextes. Ainsi, Broda *et al.* (2009) proposent de pondérer les contextes non pas en utilisant les fréquences des contextes à l'état brut comme il est d'usage, mais en ordonnant les contextes en fonction de leur fréquence, puis se servent du rang pour pondérer les contextes. D'autres approches s'appuient sur des modèles de langue pour déterminer les substituts les plus probables pour représenter les contextes (Baskaya *et al.*, 2013). Ces modèles assignent des probabilités à des séquences arbitraires de mots en se basant sur les fréquences de co-occurrence dans un corpus d'entraînement (Yuret, 2012). Ces mots substituts et leurs probabilités sont ensuite utilisés pour créer des paires de mots de manière à alimenter un modèle de co-occurrence, avant d'utiliser un algorithme de clustering. Ces méthodes sont limitées car leur performance est proportionnelle à la taille du vocabulaire et elles nécessitent de disposer de données d'entraînement importantes.

L'influence sur les contextes peut également être réalisée en y intégrant de l'information sémantique supplémentaire. En effet, il a été démontré que l'intégration de ce type d'information afin de modifier la mise en œuvre classique de l'AD permet d'améliorer sa performance (Tsatsaronis & Panagiotopoulou, 2009). Cette information sémantique, ou plus précisément les relations sémantiques, peuvent être calculées automatiquement ou provenir d'une ressource existante. Ainsi, avec un amorçage, Zhitomirsky-Geffet & Dagan (2009) modifient les poids des éléments au sein des contextes en se basant sur les voisins sémantiques trouvés à l'aide d'une mesure de similarité distributionnelle. En s'appuyant sur ces travaux, Ferret (2013) s'intéresse au problème des mots de faibles fréquences. Afin de mieux prendre en compte ces informations, il propose d'utiliser un jeu d'exemples positifs et négatifs sélectionnés de manière non-supervisée à partir d'un thésaurus distributionnel, et ainsi entraîner un classifieur supervisé. Ce classifieur est ensuite appliqué pour réordonnancer les voisins sémantiques. La méthode permet ainsi d'améliorer la qualité de la relation de similarité entre des noms de faible ou moyenne fréquence.

D'autres travaux s'intéressent au problème de la dispersion des données d'un point de vue algorithmique en cherchant à limiter les dimensions de la matrice des contextes, notamment en la lissant afin de réduire le nombre de composants vectoriels (Turney & Pantel, 2010). Ainsi, l'analyse Sémantique Latente (LSA) (Landauer & Dumais, 1997; Padó & Lapata, 2007) met en œuvre une méthode de factorisation de matrice par Décomposition en Valeurs Singulières (SVD). Les données originales de la matrice des contextes sont abstraites en composants linéaires indépendants, permettant ainsi de réduire le bruit et d'en faire ressortir les éléments essentiels. Outre la réduction du coût de traitement, la réduction de dimension améliore considérablement la précision dans les applications de la LSA. Par exemple, l'application de la

SVD à la similarité entre mots permet ainsi d'atteindre des scores équivalents à ceux d'un humain dans un test du TOEFL avec des questions de synonymie à choix multiples (Landauer & Dumais, 1997). Ceci s'explique, entre autres, par le fait qu'avec les mesures de similarité les plus fréquemment employées, les termes sont vus uniquement comme similaires s'ils apparaissent dans les mêmes contextes. En ce qui concerne les mots de faible fréquence, la SVD est une manière de simuler les contextes manquants, en compensant le manque de données (Vozalis & Margaritis, 2003). Certaines méthodes, comme la factorisation en matrice non-négative (Lee & Seung, 1999), permettent de mieux modéliser la fréquence des mots. Mais, lorsqu'il s'agit d'acquérir des relations sémantiques, les performances semblent moins bonnes que celles obtenues avec la LSA (Turney & Pantel, 2010; Utsumi, 2010).

Aussi, la réduction de dimensions facilite le traitement des vecteurs de contextes, mais ne résout pas le problème initial de construction d'une matrice de co-occurrence potentiellement immense. Ainsi, l'indexation aléatoire, ou *Random Indexing* (RI) Kanerva *et al.* (2000), apporte une solution à ce problème en construisant incrémentalement la matrice des contextes en fonction d'un vecteur d'index du mot cible généré aléatoirement. Cette approche permet d'éviter la construction d'une matrice trop grande tout en réduisant la dimension de la matrice. Les performances obtenues avec le RI sont alors équivalentes à celles obtenues avec la LSA lors de l'identification de synonymes de manière similaire au test du TOEFL (Karlgren & Sahlgren, 2001). Récemment, Polajnar & Clark (2014) ont montré que la sélection des meilleurs contextes combinés à une normalisation de leur poids permet d'améliorer la qualité de la matrice obtenue par SVD. Dans des cadres applicatifs comme la recherche de définition et le calcul de similarité entre syntagmes, leur impact sur les performances de modèles de sémantique compositionnelle dépend des opérateurs utilisés.

A l'instar de (Tsatsaronis & Panagiotopoulou, 2009; Ferret, 2013), notre approche ajoute des informations sémantiques dans les contextes distributionnels, mais notre objectif diffère : il s'agit de réduire le nombre de contextes et d'augmenter leur fréquence. Et contrairement aux méthodes basées sur la SVD qui limitent les contextes en supprimant de l'information, les contextes sont regroupés en généralisant l'information en contexte grâce à l'intégration de connaissances sémantiques supplémentaires calculées sur le corpus de travail.

3 Matériel

Nous présentons dans cette section le corpus de travail ainsi que les approches mises en œuvre pour acquérir les relations sémantiques utilisées lors de la généralisation des contextes.

3.1 Corpus

Pour évaluer notre approche, nous avons utilisé le corpus Menelas (Zweigenbaum, 1994). Il s'agit d'une collection de textes du domaine médical, en français, dont la thématique est les maladies coronariennes. Le corpus comporte 84 839 mots. Il est constitué de deux grandes parties : un manuel de référence sur la coronarographie et les maladies coronariennes (environ 15 000 mots), et un ensemble de comptes rendus d'hospitalisation et de lettres de médecins hospitaliers aux médecins traitants concernant des malades atteints d'une maladie coronarienne (environ 70 000 mots).

Le corpus a été analysé à travers la plate-forme de TAL Ogmios (Hamon *et al.*, 2007). Nous avons configuré la plate-forme de manière à ce que cette analyse linguistique comprenne un étiquetage morpho-syntaxique et une lemmatisation du corpus, à l'aide de TreeTagger (Schmid, 1994), et une extraction de termes à l'aide de YaTeA (Aubin & Hamon, 2006), celle-ci permettant d'identifier dans notre corpus de travail, les groupes nominaux dénotant les notions du domaine.

3.2 Acquisition de relations sémantiques

La méthode de généralisation des contextes distributionnels s'appuie sur des relations sémantiques existantes. Pour obtenir ces relations à partir de corpus, nous avons choisi d'utiliser plusieurs approches classiques d'acquisition de relations sémantiques entre termes : des patrons lexico-syntaxiques (PLS) dédiés à l'acquisition de relations d'hyperonymie, l'hypothèse d'inclusion lexicale (IL), et des règles de variation terminologique (VT).

Patrons lexico-syntaxiques (PLS) Nous utilisons les patrons définis par (Morin & Jacquemin, 2004) pour acquérir des relations d'hyperonymie entre termes simples ou complexes, soit par exemple :

— {quelques | plusieurs etc.} SN : LISTE.
— {autre} ? SN tels que LISTE.
— ...

où SN est un syntagme nominal et LISTE une liste de syntagmes.

Inclusion lexicale (IL) Cette approche s'appuie sur l'hypothèse selon laquelle si un terme (ex : *infarctus*) est inclus lexicalement dans un autre (ex : *infarctus du myocarde*) il existe généralement une relation d'hyperonymie entre ces deux termes (Grabar & Zweigenbaum, 2002). Nous contraignons l'approche en exploitant l'analyse syntaxique des termes fournie par YATEA . Nous ne considérons ici que les relations syntaxiques entre le terme complexe et sa tête.

Variation terminologique (VT) Nous utilisons la méthode d'acquisition de variantes terminologiques proposés par (Jacquemin, 2001) et implémentée dans Faster. Cette méthode exploite des règles de transformation morpho-syntaxique décrivant la variation terminologique. Sur notre corpus, il s'agit essentiellement de règles d'insertion (*chirurgie coronarienne / chirurgie de revascularisation coronarienne*, *anomalie significative / anomalie coronarienne significative*) qui permettent d'identifier des relations d'hyperonymie entre termes complexes.

Nous disposons ainsi de trois sources de relations sémantiques offrant principalement des relations d'hyperonymie. Les patrons lexico-syntaxiques nous fournissent le moins de relations, avec 98 relations d'hyperonymie. L'inclusion lexicale nous permet de disposer un nombre nettement plus important de relations : 7 187. Enfin, nous avons pu acquérir 171 variantes terminologiques.

4 Méthode de généralisation de contextes distributionnels

L'analyse distributionnelle appliquée à des corpus de spécialité ou des corpus de petite taille souffre d'une dispersion des données : la matrice des contextes, représentant la distribution des mots ou des termes, est très creuse (beaucoup d'éléments ont une valeur nulle). Une solution à ce problème consiste à densifier la matrice des contextes en faisant abstraction des variations superficielles ou des contextes peu significatifs statistiquement ou liés au bruit de la méthode d'identification de ces distributions. Pour cela, nous avons cherché, dans un premier temps, à filtrer les contextes de manière à sélectionner ceux qui semblent les plus pertinents, et surtout, à généraliser les contextes en exploitant des informations sémantiques extraites du corpus. En particulier, nous utilisons des relations sémantiques acquises automatiquement par des approches utilisées habituellement sur les corpus de spécialité : patrons lexico-syntaxiques, inclusion lexicale, variation terminologique.

Dans un premier temps, nous décrivons le processus d'analyse distributionnelle mis en œuvre, puis nous présentons la méthode de généralisation des contextes distributionnels que nous proposons.

4.1 Méthode distributionnelle

Dans le contexte d'applications en langue de spécialité, l'identification de relations sémantiques entre des noms et des termes est primordiale. Pour cela, nous nous restreignons à l'analyse distributionnelle entre des noms et des termes simples ou complexes. Ces deux catégories de mots constitueront les mots cibles.

La méthode d'analyse distributionnelle que nous avons mise en œuvre suit le schéma présenté dans la figure 1. Il est d'abord nécessaire de définir les contextes distributionnels des mots cibles. Nous avons ainsi choisi d'utiliser des fenêtres graphiques d'une largeur donnée. Ainsi, les contextes sont composés de mots qui co-occurent avec le mot cible au sein de la fenêtre graphique. Nous considérons comme contexte les adjectifs, noms, verbes et termes en écartant les mots vides (déterminants, conjonctions, adverbes, etc.). Que ce soit pour les mots cibles ou les contextes, nous considérons leurs formes lemmatisées.

Bien que les contextes soient généralement calculés sur des dépendances syntaxiques, nous avons choisi d'utiliser des contextes graphiques au sein d'une phrase et autour d'un mot cible pour plusieurs raisons :

— les textes de spécialité nécessitent une analyse particulière et nous ne disposons pas d'un analyseur syntaxique adapté.

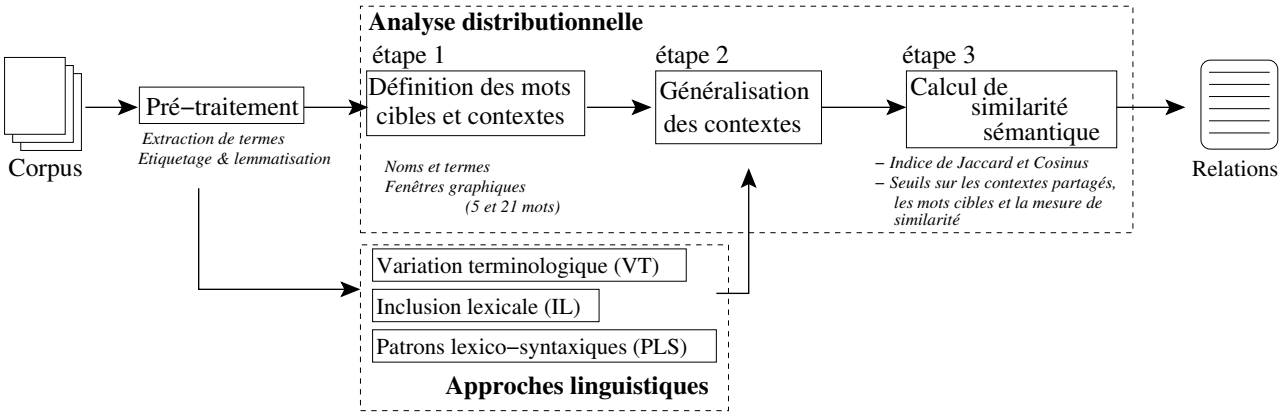


FIGURE 1 – Processus d'analyse distributionnelle

— les fenêtres graphiques étant moins restreintes que l'analyse syntaxique, elles permettent de prendre en compte un plus grand nombre de contextes, ce qui facilite la généralisation des contextes.
— l'intégration d'une analyse syntaxique dédiée à la langue générale peut nécessiter une mise en place assez coûteuse sans réellement apporter de plus value lors de l'analyse distributionnelle.

La phase de généralisation des contextes intervient après la définition des contextes distributionnels. Nous décrivons cette étape en détail à la section suivante.
Après avoir extrait et généralisé les contextes d'apparition de chaque mot cible, un score de similarité sémantique est calculé pour chaque couple de mots cibles, c'est-à-dire entre deux vecteurs de contexte Wa et Wb, où Wa_i est le ième composant du vecteur Wa. Nous avons utilisé la généralisation pondérée de l'indice de Jaccard, celle-ci étant reconnue pour être adaptée aux corpus de spécialité (Grefenstette, 1994) :

$$sim_{JACCARD}(Wa, Wb) = \frac{\sum_{i=1}^{n} min(Wa_i, Wb_i)}{\sum_{i=1}^{n} max(Wa_i, Wb_i)}$$

Nous utilisons également le cosinus. Cette mesure reflète l'angle entre deux vecteurs représentant chacun un mot cible :

$$sim_{COSINUS}(Wa, Wb) = \frac{Wa \cdot Wb}{||Wa||\ ||Wb||}$$

Le score de similarité permet ainsi de quantifier dans quelle proportion deux mots cibles sont proches. Il est cependant nécessaire d'appliquer un filtrage afin de limiter le nombre de relations proposé et d'écarter les relations potentiellement fausses. Pour cela, il est possible d'appliquer un seuil sur le score de similarité et afin de retenir que les relations dont la similarité est suffisamment élevée. Nous avons également cherché à densifier la matrice des contextes en appliquant des seuils sur trois paramètres distributionnels : le nombre de contextes partagés, la fréquence des contextes partagés, la fréquence des mots cibles. Pour chaque paramètre, un seuil est calculé automatiquement, et correspond à la moyenne des valeurs prises par chaque paramètre sur l'ensemble du corpus (après expérimentation de la médiane et de la moyenne). Lors des expériences présentées en section 5, nous testerons l'impact de ces seuils sur les résultats.

4.2 Règles de généralisation des contextes distributionnels

Le processus de généralisation et de normalisation des contextes intervient après l'étape de définition de ces contextes. L'objectif est d'une part, de diminuer la diversité des contextes distributionnels, d'autre part d'augmenter le nombre d'occurrences des contextes, c'est-à-dire leur fréquence. Ces contextes résultent de l'application de règles de généralisation et de normalisation. Dans cette perspective, nous avons choisi d'utiliser des informations sémantiques additionnelles, calculées sur le corpus, et fournissant des indices de généralisation.

Ainsi, une fois que les mots cibles et les contextes ont été définis, nous généralisons les contextes avec des relations sémantiques acquises automatiquement sur le corpus de travail à l'aide des méthodes décrites à la section 3.2 : patrons

lexico-syntaxiques dédiés à l'hyperonymie, inclusion lexicale, variation terminologique. Les deux premières méthodes proposent en général principalement des relations d'hyperonymie et seront utilisées pour généraliser les contextes. En revanche, la variation terminologique ne propose pas de relations typées sémantiquement. Aussi, étant donné que l'opération d'insertion est la seule utilisée pour acquérir des variantes, nous avons considéré que les relations obtenues étaient des relations d'hyperonymie. Le terme hyperonyme et le terme hyponyme sont identifiés à partir du nombre de mots présents dans chaque terme : le terme le plus court correspond alors à l'hyperonyme (*lésion significative*), et le terme le plus long à l'hyponyme (*lésion coronaire significative*).

Nous disposons alors, pour chaque mot w_i apparaissant dans le contexte du mot cible w, de trois ensembles de relations d'hyperonymie, $\mathbb{H}_s(w_i) = \{H_1, \ldots, H_n\} : \mathbb{H}_{PLS}$, $\mathbb{H}_{IL}$ et $\mathbb{H}_{VT}$, l'ensemble des hyperonymes pouvant être vide. Nous avons défini deux règles de substitution permettant de généraliser les contextes. Ainsi, pour chaque mot w_i dans le contexte d'un mot w, nous appliquons l'une des règles suivantes :

1. si $|\mathbb{H}_S(w_i)| = 1$, alors $w_i := H_1$
   Si au mot dans le contexte correspond un seul hyperonyme (H_1) acquis par une ou plusieurs méthodes S, le mot est remplacé par cet hyperonyme. Par exemple, si l'inclusion lexicale fournit la relation *restriction / restriction du débit coronaire*, *restriction du débit coronaire* est remplacée par *restriction*.
2. si $|\mathbb{H}_S(w_i)| > 1$, $w_i = \text{argmax}_{|H_i|}(\mathbb{H}_S(w_i))$
   Si le contexte correspond à plusieurs hyperonymes acquis par une ou plusieurs méthodes S, nous prenons en compte la fréquence des hyperonymes $|H_1|, \ldots, |H_n|$ dans le corpus, et nous choisissons l'hyperonyme dont la fréquence est la plus élevée dans le corpus.
   Par exemple, si pour le mot *artère coronaire* dans le contexte les patrons lexico-syntaxiques fournissent les hyperonymes suivants : *veine*, *artère*, *vaisseau*, celui qui est le plus fréquent est choisi et utilisé pour remplacer *artère coronaire* dans le contexte.

Quand plusieurs ensembles de relations d'hyperonymie sont disponibles, la phase de généralisation des contextes est réalisée individuellement ou de manière séquentielle : les contextes sont généralisés en utilisant les ensembles de relations les uns à la suite des autres.

5 Expériences et évaluation

5.1 Expériences

Nous avons réalisé plusieurs séries d'expériences sur le corpus Menelas afin d'évaluer l'impact des règles de généralisation proposées. Nous utilisons comme résultats de référence (*baseline*) les résultats obtenus avec l'analyse distributionnelle seule. Nous avons tout d'abord évalué l'importance des seuils sur les paramètres distributionnels (voir section 4.1). Chaque expérience décrite ci-dessous a été réalisée en appliquant ou non les seuils définis. Lorsque les seuils sont utilisés, ceux-ci ont été calculés sur la *baseline*. La table 1 résume les valeurs des seuils utilisées.

Paramètres	Fenêtre de 21 mots	Fenêtre de 5 mots
Score de similarité	Jaccard : $sim > 0,000999$ Cosinus : $sim > 0.9699$	Jaccard : $sim > 0,000999$ Cosinus : $sim > 0.9699$
Nombre de contextes partagés	2	1
Fréquence des contextes	3	2
Fréquence des mots cibles	3	3

TABLE 1 – Définition des valeurs des seuils sur les paramètres distributionnels et sur le score de similarité, en fonction de la taille des fenêtres (de 21 et 5 mots) et des mesures de similarité (Jaccard et Cosinus)

Afin de cerner la contribution de chaque méthode linguistique décrite à la section 3.2, nous avons défini un ensemble d'expériences où la généralisation des contextes est réalisée en utilisant les relations d'hyperonymie proposées par chaque méthode individuellement. Les règles de généralisation des contextes distributionnels w_i sont alors appliquées en utilisant séparément les ensembles $\mathbb{H}_{PLS}(w_i)$ – relations d'hyperonymie acquises à l'aide des patrons lexico-syntaxiques

(AD/PLS), $\mathbb{H}_{IL}(w_i)$ – relations d'hyperonymie issues de l'inclusion lexicale (AD/IL), et $\mathbb{H}_{VT}(w_i)$ – variantes terminologiques (AD/VT).

Puis, de manière séquentielle, nous avons appliqué les règles de généralisation en utilisant les ensembles de relations d'hyperonymie proposées par deux approches linguistiques ($\mathbb{H}_{PLS}(w_i)$ puis $\mathbb{H}_{IL}(w_i)$ – AD/PLS+IL, $\mathbb{H}_{VT}(w_i)$ puis $\mathbb{H}_{PLS}(w_i)$ – AD/VT+PLS, etc.). Tous les contextes sont alors généralisés en utilisant les relations proposées par l'un des ensembles (par exemple $\mathbb{H}_{PLS}(w_i)$), puis les contextes généralisés ou non sont à nouveau généralisés en utilisant un autre ensemble de relations (par exemple $\mathbb{H}_{IL}(w_i)$). De même, nous combinons les trois ensembles de relations (par exemple, $\mathbb{H}_{PLS}(w_i)$ puis $\mathbb{H}_{IL}(w_i)$ puis $\mathbb{H}_{VT}(w_i)$ – AD/PLS+IL+VT). En combinant des sources de relations d'hyperonymie de plusieurs manières, nous souhaitons d'une part, évaluer la complémentarité des approches utilisées pour généraliser les contextes, et d'autre part, étudier l'impact de l'ordre de ces méthodes dans la séquence de généralisation.

Nous avons également considéré toutes les relations d'hyperonymie indépendamment de la méthode utilisée pour les acquérir. On considère alors l'ensemble $H(w_i) = \mathbb{H}_{PLS}(w_i) \cup \mathbb{H}_{IL}(w_i) \cup \mathbb{H}_{VT}(w_i)$ – AD/ALL3, pour appliquer les règles de généralisation sur le contexte w_i.

L'ensemble des expériences a été réalisé sur deux tailles de fenêtres : 5 mots ($\pm$ 2 mots, centrée sur le mot cible) et 21 mots ($\pm$ 10 mots, centrée sur le mot cible). En effet, la taille des fenêtres a une influence sur le nombre et la qualité mais aussi sur le type des relations obtenues par analyse distributionnelle. En général, une fenêtre de taille restreinte (5 mots) permet de disposer d'un plus grand nombre de contextes pertinents pour un mot cible donné, mais conduit à une dispersion des données plus important qu'avec une fenêtre plus large (Rapp, 2003). De plus, cette influence peut aussi dépendre de la taille du corpus : des fenêtres plus grandes peuvent être intéressantes pour des corpus plus petits (Kiela & Clark, 2014). De plus, les résultats obtenus avec des fenêtres de taille restreinte sont de meilleure qualité, en particulier pour des relations classiques (synonymie, antonymie, hyperonymie, méronymie, etc.) alors que des fenêtres plus larges sont plus adaptées à l'identification de relations spécifiques au domaine (Sahlgren, 2006; Peirsman *et al.*, 2008).

5.2 Évaluation

La qualité des résultats obtenus lors de nos expériences est évaluée en comparant les relations sémantiques acquises aux 1 735 419 relations fournies par la partie française de l'UMLS [1], soit 2 434 relations entre les termes correspondant au vocabulaire du corpus. Ces relations sont des hyperonymes (muscle papillaire/cœur), des co-hyponymes (vertige/douleur, œsophagite/gastrite), des synonymes (chute/baisse) et des relations du domaine (angiographie/artère, souffle cardiaque/douleur). À l'instar de (Curran, 2004) et (Ferret, 2013), nous considérons ici les relations obtenues comme des ensembles de voisins associés à des mots cibles, les voisins étant ordonnés suivant la similarité avec le mot cible.

L'évaluation des résultats est réalisée avec des mesures d'évaluation utilisées habituellement sur les résultats d'une analyse distributionnelle : la macro-précision (Sebastiani, 2002), la moyenne des précisions moyennes (MAP) (Buckley & Voorhees, 2005) et la R-précision.

La macro-précision est la moyenne des précisions $p(w_i)$ obtenues pour chaque mot cible (w_i) et un ensemble de voisins sémantiques I_i^j ($I_i^{j(+)}$ étant un voisin pertinent pour le mot cible considéré, et n_i le nombre de voisins considérés) :

$$p(w_i) = \frac{\sum_{j=1}^{n_i} I_i^{j(+)}}{\sum_{j=1}^{n_i} I_i^j}$$

La macro-précision pour l'ensemble des mots cibles est alors : $P = \frac{\sum_{k=1}^{|w_i|} p(w_k)}{|w_i|}$

Nous considérons quatre sous-ensembles voisins permettant d'obtenir la macro-précision après examen de 1 ($n_i = 1$, P@1), 5 ($n_i = 5$, P@5), 10 ($n_i = 10$, P@10) et 100 voisins ($n_i = 100$, P@100) :

$$P@N = \sum_{i=1}^{|w_i|} p(w_i | n_i = N)$$

La macro-précision permet d'évaluer une qualité globale des résultats tout en considérant que tous les mots cibles ont le même poids quel que soit le nombre de voisins, alors que la micro-précision aurait tendance à privilégier les mots cibles

1. http://www.nlm.nih.gov/research/umls/

comportant beaucoup de voisins, dont une bonne partie ne sont probablement pas pertinents, au détriment de mots cibles ayant peu de voisins. Pour P@1, la macro-précision est équivalente à la micro-précision.

Une alternative consiste à utiliser comme seuil n_i le nombre de voisins corrects attendus pour un mot cible. Pour cela, nous avons utilisé la R-précision (Buckley & Voorhees, 2005).

Pour le calcul de la R-précision, nous comparons nos résultats non plus à l'ensemble des relations de la partie française de l'UMLS, mais à des ensembles de référence constitués à partir de cette ressource. Il s'agit de réduire les relations de référence aux seules relations entre des termes ou des mots présents dans le corpus de travail et dans chaque expérience. Ainsi, nous disposons d'autant de références que d'expériences, avec par exemple entre 24 et 46 relations pour les expériences avec une fenêtre de 21 mots et la mesure de Cosinus.

La moyenne des précisions moyennes (MAP) est obtenue en considérant la précision non interpolée $UAP(I_i^j)$ des voisins sémantiques I_i^j au rang j, n_i est le nombre de voisins sémantiques I_i^j du mot cible w_i. La MAP est alors la moyenne de ces précisions non interpolées :

$$MAP = \frac{1}{|w_i|} \sum_{i=1}^{|w_i|} \frac{1}{n_i} \sum_{j=1}^{n_i} UAP(I_i^j)$$

La MAP est le reflet de la qualité du classement et permet d'évaluer la pertinence de la mesure de similarité utilisée. Ainsi, elle valorise le fait que la méthode ordonne tous les voisins sémantiques corrects proches de la tête de liste. Réciproquement, le fait d'ajouter des voisins sémantiques incorrects en fin de liste (après les voisins corrects) ne pénalise pas la méthode.

6 Résultats et discussion

Dans cette section, nous présentons et discutons les résultats obtenus tout d'abord avec une fenêtre de 5 mots puis avec celle de 21 mots. Les résultats de la mesure Cosinus sont discutés mais ne sont pas présentés dans les tableaux.

	Sans seuil						Avec seuil					
	Rel. acq.	Rel. UMLS	MAP	R-préc	P@1	P@5	Rel. acq.	Rel. UMLS	MAP	R-préc	P@1	P@5
ADSeule	34132	98	0,172	0,057	0,098	0,043	1342	24	0,496	0,375	0,375	0,125
AD/VT	25578	84	0,158	0,055	0,068	0,050	1402	24	0,487	0,333	0,333	0,133
AD/IL	11658	56	0,164	0,066	0,079	0,047	694	12	0,443	0,333	0,333	0,100
AD/PLS	23760	84	0,181	0,100	0,114	0,050	1252	20	0,570	0,500	0,500	0,133
AD/VT+IL	12030	56	0,161	0,066	0,079	0,047	696	12	0,443	0,333	0,333	0,100
AD/VT+PLS	22176	82	0,185	0,102	0,116	0,051	1188	20	0,570	0,500	0,500	0,133
AD/IL+VT	11434	54	0,158	0,069	0,083	0,044	660	12	0,525	0,500	0,500	0,100
AD/IL+PLS	10456	52	0,177	0,100	0,114	0,046	610	10	0,525	0,500	0,500	0,100
AD/PLS+VT	22176	82	0,185	0,102	0,116	0,051	1188	20	0,570	0,500	0,500	0,133
AD/PLS+IL	11280	56	0,166	0,066	0,079	0,047	688	10	0,443	0,333	0,333	0,100
AD/VT+IL+PLS	10808	52	0,174	0,100	0,114	0,046	616	10	0,525	0,500	0,500	0,100
AD/VT+PLS+IL	11642	56	0,162	0,066	0,079	0,047	694	10	0,443	0,333	0,333	0,100
AD/IL+VT+PLS	5608	34	0,212	0,100	0,100	0,067	912	10	0,229	0,000	0,500	0,100
AD/IL+PLS+VT	10244	52	0,178	0,100	0,114	0,046	576	10	0,526	0,500	0,500	0,100
AD/PLS+VT+IL	6392	42	0,274	0,162	0,177	0,071	1036	10	0,585	0,500	0,250	0,133
AD/PLS+IL+VT	6020	40	0,233	0,109	0,125	0,063	970	10	0,381	0,250	0,222	0,100
AD/ALL3	11266	56	0,148	0,066	0,079	0,047	694	6	0,4431	0,333	0,333	0,100

TABLE 2 – Résultats obtenus avec la mesure de Jaccard, évalués avec la MAP, R-précision, et précision à 1 et 5 pour une fenêtre de 5 mots – sans et avec seuil sur la mesure de similarité

6.1 Fenêtre graphique restreinte

Pour une fenêtre de 5 mots, nous présentons uniquement les résultats obtenus en utilisant la mesure de Jaccard, sans et avec seuils (tableau 2). La mesure du cosinus donne de très faibles résultats quels que soient les paramètres et ne semble

pas adaptée aux fenêtres de petite taille et lorsque les fréquences sont faibles : la précision (P@1) des résultats obtenus avec le cosinus varie entre 0,02 et 0,06, alors que pour la mesure de Jaccard, nous obtenons des précisions variant entre 0,01 et 0,17. Lorsqu'aucun seuil n'est appliqué sur les paramètres distributionnels, nous constatons que la généralisation à l'aide des relations acquises grâce aux patrons lexico-syntaxiques améliore la qualité des résultats aussi bien en termes de précision ou de R-précision que de MAP. De plus, la variation terminologique tend à dégrader les résultats quand elle est utilisée individuellement alors qu'elle a un impact positif lorsqu'elle est combinée aux relations issues des patrons lexico-syntaxiques. Mais l'apport des relations issues des patrons lexico-syntaxiques est annulé quand on ajoute celles issues de l'inclusion lexicale. Nous pouvons également faire ce constat lorsque l'on considère les relations indépendamment de la méthode utilisée pour les produire. La généralisation des contextes est alors peut-être trop importante pour pouvoir être utile dans l'analyse distributionnelle. Aussi, l'analyse des variations de la MAP permettent également de constater que la généralisation des contextes améliore le classement des relations présentes dans l'UMLS.

A la vue des mesures, la qualité des résultats semble bénéficier de la combinaison des trois sources de relations d'hyperonymie. De même, le nombre de relations retournées est réduit au minimum d'un quart avec la généralisation, par rapport à l'ensemble de relations obtenues avec l'AD seule, voire divisé par 6 pour certaines combinaison comme PLS+VT+IL. Cependant, le nombre de relations retrouvées dans l'UMLS est divisé par deux (98 avec l'AD seule, 42 avec la combinaison offrant la meilleure précision) lorsque nous utilisons les relations proposées par plusieurs méthodes. Si des constats similaires peuvent être réalisés lorsque nous appliquons des seuils sur les paramètres distributionnels, nous remarquons également l'impact positif des relations acquises par inclusion lexicale sur les résultats. Mais ici, les résultats sont probablement peu significatifs. En effet, peu de relations sont retrouvées dans l'UMLS lorsque nous utilisons Jaccard pour mesure la similarité entre les mots, et nous pouvons douter de la significativité statistique des résultats obtenus avec les seuils. Une évaluation manuelle est nécessaire.

6.2 Fenêtre graphique large

Lorsqu'une fenêtre large de 21 mots est utilisée, les observations et les résultats sont différents selon les mesures de similarité utilisées et l'utilisation (cf. tableau 4) ou non (cf. tableau 3) de seuils sur les paramètres distributionnels. Ainsi avec la mesure de Jaccard, la qualité des résultats est améliorée si les contextes sont généralisés avec des relations issues des patrons lexico-syntaxiques et qu'aucun seuil n'est appliqué. La contribution des relations acquises par inclusion lexicale est variable : l'utilisation de ces relations pour généraliser les contextes dégradent les résultats si l'on n'applique pas de seuil sur les paramètres distributionnels, et au contraire permettent d'obtenir les meilleurs résultats lorsque des seuils sont utilisés. L'impact des patrons lexico-syntaxiques seul est beaucoup plus faible avec le cosinus. Mais lorsqu'ils sont pris en compte après l'utilisation des relations acquises par inclusion lexicale, la précision est améliorée et même supérieure à celle obtenue avec Jaccard. De plus, nous pouvons noter que les variantes terminologiques ont un impact nul ou négatif sur la qualité des résultats.

	Rel. acquises		Rel. dans UMLS		MAP		R-précision		P@1		P@5	
	JACC	COS	JACC	COS	JACC	COS	JACC	COS	JACC	COS	JACC	COS
ADSeule	9256	9256	46	46	0,221	0,149	0,142	0,098	0,118	0,088	0,059	0,028
AD/VT	8758	8758	44	44	0,201	0,158	0,120	0,104	0,094	0,094	0,056	0,053
AD/IL	6360	6360	42	42	0,197	0,120	0,075	0,081	0,097	0,065	0,071	0,056
AD/PLS	8418	8418	42	42	0,243	0,165	0,172	0,111	0,133	0,100	0,080	0,026
AD/VT+IL	6312	6312	42	42	0,196	0,120	0,075	0,081	0,097	0,065	0,077	0,060
AD/VT+PLS	7972	7972	42	42	0,244	0,166	0,172	0,111	0,133	0,100	0,080	0,026
AD/IL+VT	6138	6138	40	40	0,175	0,128	0,046	0,086	0,069	0,069	0,069	0,060
AD/IL+PLS	5874	5874	40	40	0,201	0,191	0,046	0,155	0,069	0,138	0,083	0,028
AD/PLS+VT	7972	7972	42	42	0,244	0,166	0,172	0,111	0,133	0,100	0,080	0,041
AD/PLS+IL	6346	6346	42	42	0,220	0,116	0,108	0,065	0,129	0,065	0,084	0,060
AD/VT+IL+PLS	5828	5828	40	40	0,198	0,191	0,046	0,155	0,069	0,138	0,076	0,026
AD/VT+PLS+IL	6310	6310	42	42	0,219	0,115	0,108	0,065	0,129	0,065	0,077	0,041
AD/IL+VT+PLS	5662	5662	40	40	0,202	0,193	0,046	0,155	0,069	0,138	0,083	0,026
AD/IL+PLS+VT	5662	5662	40	40	0,202	0,193	0,046	0,155	0,069	0,138	0,083	0,041
AD/PLS+VT+IL	6310	6310	42	42	0,219	0,115	0,108	0,065	0,129	0,065	0,077	0,041
AD/PLS+IL+VT	6122	6122	40	40	0,199	0,123	0,081	0,069	0,103	0,069	0,083	0,026
AD/ALL3	6306	6306	42	42	0,222	0,120	0,108	0,081	0,129	0,065	0,084	0,026

TABLE 3 – Résultats évalués avec la MAP, R-précision, et précision à 1 et 5 pour une fenêtre de 21 mots – sans seuil sur la similarité sémantique

	Rel. acquises		Rel. dans UMLS		MAP		R-précision		P@1		P@5	
	JACC	COS	JACC	COS	JACC	COS	JACC	COS	JACC	COS	JACC	COS
ADSeule	392	6960	4	40	0,406	0,169	0,250	0,100	0,250	0,100	0,1000	0,0600
AD/VT	418	6576	6	38	0,181	0,180	0,000	0,107	0,000	0,107	0,040	0,064
AD/IL	370	4580	4	34	0,532	0,141	0,000	0,077	0,500	0,077	0,100	0,031
AD/PLS	390	6262	6	36	0,219	0,190	0,500	0,115	0,000	0,115	0,120	0,069
AD/VT+IL	380	4568	4	34	0,531	0,141	0,500	0,077	0,500	0,077	0,100	0,031
AD/VT+PLS	354	5910	6	36	0,220	0,190	0,000	0,115	0,000	0,115	0,120	0,069
AD/IL+VT	352	4414	2	32	0,533	0,152	0,500	0,083	0,500	0,083	0,100	0,033
AD/IL+PLS	334	4216	4	32	0,371	0,228	0,250	0,167	0,250	0,167	0,100	0,050
AD/PLS+VT	354	5910	6	36	0,220	0,190	0,000	0,115	0,000	0,115	0,120	0,069
AD/PLS+IL	404	4514	6	34	0,428	0,134	0,333	0,077	0,333	0,077	0,133	0,031
AD/VT+IL+PLS	338	4208	4	32	0,371	0,228	0,250	0,167	0,250	0,167	0,100	0,050
AD/VT+PLS+IL	404	4514	6	34	0,428	0,133	0,333	0,077	0,333	0,077	0,133	0,031
AD/IL+VT+PLS	316	4056	4	32	0,372	0,230	0,250	0,167	0,250	0,167	0,100	0,050
AD/IL+PLS+VT	314	4056	4	32	0,372	0,230	0,250	0,167	0,250	0,167	0,100	0,050
AD/PLS+VT+IL	404	4514	6	34	0,428	0,133	0,333	0,077	0,333	0,077	0,133	0,031
AD/PLS+IL+VT	378	4352	4	32	0,376	0,144	0,250	0,083	0,250	0,083	0,150	0,033
AD/ALL3	380	4544	6	34	0,430	0,140	0,333	0,077	0,333	0,077	0,133	0,031

TABLE 4 – Résultats évalués avec la MAP, R-précision, et précision à 1 et 5 pour une fenêtre de 21 mots – avec seuils sur la similarité sémantique

En ce qui concerne la précision P@5, bien que les valeurs soient plus faibles que la précision P@1, nous observons que l'inclusion permet d'augmenter les valeurs de précision, en particulier lorsque celle-ci est utilisée après les patrons lexico-syntaxiques. Aussi, en terme de R-précision, les résultats sont plus contrastés qu'avec P@1 : pour les meilleures configurations de généralisation des contextes, les valeurs de R-précision sont identiques ou supérieures. Les relations attendues, sont classées parmi les premières ou à un rang plus élevé qu'avec l'AD seule. En termes de MAP, les valeurs sont assez stables. Les améliorations obtenues suivant les combinaisons montrent qu'il y a plus de relations retrouvées dans l'UMLS parmi les premiers voisins. Mais la variation restant faible entre l'AD seule et l'AD avec des contextes généralisés, l'ordonnancement des voisins n'est pas beaucoup modifié par notre méthode.

Lorsque l'on considère le nombre de relations retournées par l'analyse distributionnelle, nous constatons que quelle que soit la mesure utilisée, la combinaison des trois sources de relations, de manière séquentielle ou globalement, n'a qu'un impact moyen sur l'amélioration de la qualité des résultats. Toutefois, exploiter l'ensemble des relations d'hyperonymie à notre disposition permet de réduire d'un tiers le nombre de relations retournées par l'analyse distributionnelle, tout en conservant le même nombre de relations présentes dans l'UMLS.

Comme lors de l'utilisation d'une fenêtre de taille restreinte, l'application de seuils sur les paramètres distributionnels améliore la qualité des résultats. Mais le faible nombre de relations présentes dans l'UMLS, parmi l'ensemble des relations obtenues avec Jaccard, rend les observations difficiles à interpréter. En revanche, nous pouvons noter que les résultats obtenus avec le cosinus sont meilleurs en appliquant des seuils tout en ne réduisant pas trop le nombre de relations correctes.

7 Conclusion

Nous nous sommes intéressés dans cet article à la réduction de la dispersion des données dans les matrices de vecteurs de contexte utilisés pour mettre en œuvre l'analyse distributionnelle. Pour cela, nous avons proposé une méthode de généralisation des contextes distributionnels s'appuyant sur des relations d'hyperonymie acquises en corpus. Les mots décrivant le contexte distributionnel d'un mot cible sont considérés comme des hyponymes et sont substitués par des hyperonymes identifiés sur le corpus. Nous avons réalisé un certain nombre d'expériences sur un corpus médical en combinant plusieurs paramètres. Les relations d'hyperonymie ont été acquises avec des approches habituellement utilisées sur des textes de spécialité. Bien que l'évaluation des méthodes distributionnelles soit complexe à réaliser, nous avons confronté les résultats aux relations sémantiques proposées par l'UMLS français. Plusieurs mesures d'évaluation ont été utilisées pour évaluer l'impact de la généralisation des contextes sur l'analyse distributionnelle. L'analyse des résultats montre que lorsque la taille des fenêtres graphiques permettant de produire les contextes distributionnels est petite et que l'indice de Jaccard est utilisé comme mesure de similarité, il est préférable d'utiliser les relations proposées par les patrons lexico-syntaxiques pour généraliser les contextes. Il est alors possible d'obtenir un bon compromis permettant d'avoir à

la fois une bonne couverture et une amélioration de la précision. En revanche, lorsque la taille de la fenêtre est large, la généralisation des contextes grâce aux relations issues de l'inclusion lexicale améliore les résultats si le cosinus est utilisé comme mesure de similarité.

Outre une analyse manuelle des relations et de l'impact du processus de généralisation sur les données manipulées, ces résultats ouvrent plusieurs perspectives. Les relations d'hyperonymie que nous avons utilisées ont été exploitées séparément. Or, celles-ci pourraient être considérées comme une ébauche de taxonomie et nous envisageons d'adapter la méthode de généralisation des contextes afin qu'elle prenne en compte ce réseau de relations acquises en corpus. Aussi, l'ensemble des relations acquises en corpus pouvant être bruité, nous envisageons d'utiliser d'autres sources de relations comme celles proposées par des terminologies. Il sera alors possible d'évaluer l'impact de la généralisation et de relations lorsque leur statut terminologique est maîtrisé. Enfin, la réduction de dimensions peut également s'appuyer sur un processus de normalisation des contextes. Il est alors nécessaire de prendre en compte des relations de synonymie ou d'antonymie qui peuvent également être acquises en corpus ou issues de ressources terminologiques existantes.

Références

AUBIN S. & HAMON T. (2006). Improving term extraction with terminological resources. In *Advances in Natural Language Processing*, number 4139 in LNAI, p. 380–387 : Springer.

BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, **43**(3), 209–226.

BASKAYA O., SERT E., CIRIK V. & YURET D. (2013). Ai-ku : Using substitute vectors and co-occurrence modeling for word sense induction and disambiguation. In *Proceedings of SemEval - 2013*, p. 300–306, Atlanta, Georgia, USA : Association for Computational Linguistics.

BRODA B., PIASECKI M. & SZPAKOWICZ S. (2009). Rank-based transformation in measuring semantic relatedness. In Y. GAO & N. JAPKOWICZ, Eds., *Canadian Conference on AI*, volume 5549, p. 187–190 : Springer.

BUCKLEY C. & VOORHEES E. (2005). Retrieval system evaluation. In E. VOORHEES & D. HARMAN, Eds., *TREC : Experiment and Evaluation in Information Retrieval*, chapter 3. MIT Press.

CHATTERJEE N. & MOHAN S. (2008). Discovering word senses from text using random indexing. In *Proceedings of the 9th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing'08, p. 299–310, Berlin, Heidelberg : Springer-Verlag.

CURRAN J. R. (2004). *From distributional to semantic similarity*. PhD thesis, Institute for Communicating and Collaborative Systems School of Informatics University of Edinburgh.

FERRET O. (2013). Sélection non supervisée de relations sémantiques pour améliorer un thésaurus distributionnel. In *TALN 2013*, p. 48–61, Les Sables d'Olonne, France.

FIRTH J. (1957). A synopsis of linguistic theory 1930-1955. *Studies in linguistic analysis*, p. 1–32.

GRABAR N. & ZWEIGENBAUM P. (2002). Lexically-based terminology structuring : Some inherent limits. In L.-F. CHIEN, B. DAILLE, K. KAGEURA & H. NAKAGAWA, Eds., *Proceedings of Second International Workshop on Computational Terminology (COMPUTERM 2002)*, p. 36–42, Taipei, Taiwan : ACLCLP.

GREFENSTETTE G. (1994). Corpus-derived first, second and third-order word affinities. In *Sixth Euralex International Congress*, p. 279–290.

HAMON T., NAZARENKO A., POIBEAU T., AUBIN S. & DERIVIÈRE J. (2007). A robust linguistic platform for efficient and domain specific web content analysis. In *RIAO 2007*, Pittsburgh, USA.

HARRIS Z. (1954). Distributional structure. *Word*, **10**(23), 146–162.

JACQUEMIN C. (2001). *Spotting and discovering terms through natural language processing*. The MIT Press.

KANERVA P., KRISTOFERSSON J. & HOLST A. (2000). Random indexing of text samples for latent semantic analysis. In L. GLEITMAN & A. JOSH, Eds., *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, volume 1036, Erlbaum, New Jersey.

KARLGREN J. & SAHLGREN M. (2001). From words to understanding. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 294–308 : Foundations of Real-World Intelligence.

KIELA D. & CLARK S. (2014). A systematic study of semantic vector space model parameters. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) at EACL 2014*, p. 21–30, Gothenburg, Sweden.

LANDAUER T. & DUMAIS S. (1997). A solution to plato's problem : The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review ; Psychological Review*, **104**(2), 211.

LEE D. D. & SEUNG H. S. (1999). Learning the parts of objects by nonnegative matrix factorization. *Nature*, **401**, 788–791.

LUND K. & BURGESS C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instrumentation, and Computers*, **28**, 203–208.

MORIN E. & JACQUEMIN C. (2004). Automatic Acquisition and Expansion of Hypernym Links. *Computers and the Humanities*, **38**(4), 363–396.

PADÓ S. & LAPATA M. (2007). Dependency-based construction of semantic space models. *Comput. Linguist.*, **33**(2), 161–199.

PEIRSMAN Y., KRIS H. & DIRK G. (2008). Size matters. tight and loose context definitions in english word space models. In *ESSLLI Workshop on Distributional Lexical Semantics*, Hamburg, Germany.

POLAJNAR T. & CLARK S. (2014). Improving distributional semantic vectors through context selection and normalisation. In *Proceedings of EACL 2014*, p. 230–238.

RAPP R. (2003). Word sense discovery based on sense descriptor dissimilarity. In *MT Summit'2003*, p. 315–322.

SAHLGREN M. (2006). *The Word-Space Model : Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations between Words in High-Dimensional Vector Spaces*. PhD thesis, Stockholm University, Stockholm, Sweden.

SCHMID H. (1994). Probabilistic part-of-speech tagging using decision trees. In *New Methods in Language Processing*, p. 44–49, Manchester, UK.

SEBASTIANI F. (2002). Machine learning in automated text categorization. *ACM Comput. Surv.*, **34**(1), 1–47.

TSATSARONIS G. & PANAGIOTOPOULOU V. (2009). A generalized vector space model for text retrieval based on semantic relatedness. In *EACL 2009*, p. 70–78, Stroudsburg, PA, USA : Association for Computational Linguistics.

TURNEY P. D. & PANTEL P. (2010). From frequency to meaning : Vector space models of semantics. *Journal of artificial intelligence research*, **37**, 141–188.

UTSUMI A. (2010). Evaluating the performance of nonnegative matrix factorization for constructing semantic spaces : Comparison to latent semantic analysis. In *Proceedings of SMC*, p. 2893–2900 : IEEE.

VAN DER PLAS L. (2008). *Automatic lexico-semantic acquisition for question answering*. Thèse de doctorat, University of Groningen, Groningen.

VOZALIS E. & MARGARITIS K. G. (2003). Analysis of recommender systems' algorithms. In *The 6th Hellenic European Conference on Computer Mathematics & its Applications (HERCMA), Athens, Greece*.

WEEDS J. & WEIR D. (2005). Co-occurrence retrieval : A flexible framework for lexical distributional similarity. *Comput. Linguist.*, **31**(4), 439–475.

YURET D. (2012). Fastsubs : An efficient and exact procedure for finding the most likely lexical substitutes based on an n-gram language model. *IEEE Signal Process. Lett.*, **19**(11), 725–728.

ZHITOMIRSKY-GEFFET M. & DAGAN I. (2009). Bootstrapping distributional feature vector quality. *Comput. Linguist.*, **35**(3), 435–461.

ZWEIGENBAUM P. (1994). Menelas : an access system for medical records using natural language. *Computer Methods and Programs in Biomedicine*, **45**.

Extraction non supervisée de relations sémantiques lexicales *

Juliette Conrath Stergos Afantenos Nicholas Asher Philippe Muller
IRIT, Université Toulouse & CNRS, Univ. Paul Sabatier, 118 Route de Narbonne, 31062 Toulouse
{nom.prenom@irit.fr}

Résumé. Nous présentons une base de connaissances comportant des triplets de paires de verbes associés avec une relation sémantique/discursive, extraits du corpus français frWaC par une méthode s'appuyant sur la présence d'un connecteur discursif reliant deux verbes. Nous détaillons plusieurs mesures visant à évaluer la pertinence des triplets et la force d'association entre la relation sémantique/discursive et la paire de verbes. L'évaluation intrinsèque est réalisée par rapport à des annotations manuelles. Une évaluation de la couverture de la ressource est également réalisée par rapport au corpus Annodis annoté discursivement. Cette étude produit des résultats prometteurs démontrant l'utilité potentielle de notre ressource pour les tâches d'analyse discursive mais aussi des tâches de nature sémantique.

Abstract. This paper presents a knowledge base containing triples involving pairs of verbs associated with semantic or discourse relations. The relations in these triples are marked by discourse connectors between two adjacent instances of the verbs in the triple in the large French corpus, frWaC. We detail several measures that evaluate the relevance of the triples and the strength of their association. We use manual annotations to evaluate our method, and also study the coverage of our ressource with respect to the discourse annotated corpus Annodis. Our positive results show the potential impact of our ressource for discourse analysis tasks as well as semantically oriented tasks.

Mots-clés : discours, sémantique, sémantique lexicale.

Keywords: discourse, semantics, lexical semantics.

1 Introduction

Les ressources lexicales relationnelles, c'est-à-dire qui répertorient les liens entre items lexicaux d'un point de vue sémantique, sont essentiellement tournées vers l'équivalence sémantique (synonymie, similarité) dans des thesaurus, éventuellement avec des relations hiérarchiques (hyperonymie ou hyponymie), à l'exemple de la référence Wordnet (Felbaum, 1998), qui inclut aussi des relations de partie à tout. Quand elles incluent des relations plus variées, par exemple dans les thesaurus distributionnels (Grefenstette, 1994), celles-ci ne sont pas typées. Les exceptions sont rares : le lexique sémantique FrameNet (Baker *et al.*, 1998) inclut des relations de causalité ou de précédence temporelle entre items désignant des événements, à l'intérieur de scénarios prototypiques, mais ces relations sont peu nombreuses et relativement négligeables par rapport au contenu de ce lexique ; la base Verbocean (Chklovski & Pantel, 2004) comporte des relations sémantiques de plusieurs types entre verbes transitifs : relations de type causal (*enablement*, par exemple *fight/win*), précédence temporelle (*marry/divorce*), similarité, antonymie, force (*wound/kill*), mais sa couverture est assez faible (environ 4000 paires de verbes dans sa version filtrée). Dans les deux cas les validations sont partielles, et ces travaux semblent avoir été laissés en attente. Les relations lexicales, notamment entre verbes, sont pourtant cruciales pour la compréhension du langage naturel, et sont utilisées par exemple dans la tâche d'inférence textuelle, où il s'agit de trouver les implications entre certains événements (Hashimoto *et al.*, 2009; Tremper & Frank, 2013), dans certaines tâches d'extraction, par exemple de relations temporelles (UzZaman *et al.*, 2013), dans l'analyse discursive en l'absence de marques explicites (Sporleder & Lascarides, 2008), ou bien encore pour le résumé automatique (Liu *et al.*, 2007). Certains travaux se sont consacrés à l'inventaire de tels liens pour des relations spécifiques : par exemple les liens causaux (Do *et al.*, 2011), les liens temporels (Chambers & Jurafsky, 2008), les liens d'implication (*entailment*) (Hashimoto *et al.*, 2009), implication et présupposition (Tremper & Frank, 2013). Le but de notre travail est l'extraction de certaines relations sémantiques qui sont primordiales pour l'analyse discursive. Par analyse discursive nous entendons l'établissement de liens entre énoncés, au-delà de la phrase, comme dans l'exemple (1), où en l'absence de marque explicite (par exemple avec un connecteur comme *donc* ou

*. Ce travail a bénéficié d'une aide de l'Agence Nationale de la Recherche portant la référence (ANR-12-CORD-0004)

ainsi), on peut déduire une causalité entre les événements par la connaissance de la sémantique des deux verbes mis en évidence.

(1) Le candidat a démontré tout son savoir-faire lors de la dernière épreuve. Le jury a été conquis.

L'analyse discursive est une tâche difficile, y compris pour l'humain. En effet, les relations rhétoriques sont fréquemment implicites et nécessitent des inférences pour être identifiées [1]. Ceci rend l'annotation de corpus fastidieuse et souvent imprécise, et peu de données annotées en relation sont ainsi disponibles à l'heure actuelle. De ce fait, les approches inductives par apprentissage ont un impact réduit, puisque celles-ci nécessitent un très grand nombre de données. De nombreux travaux tentent de pallier ces limitations en élaborant des approches faiblement supervisées, utilisant des données non annotées mais comportant des marques explicites repérables automatiquement pour retrouver des contextes typiques de certaines relations. Ces approches (Sporleder & Lascarides, 2008; Braud & Denis, 2013) s'appuient sur l'hypothèse de régularité des contextes, notamment des associations lexicales, mais de façon indirecte. A l'inverse, certaines approches (Wellner *et al.*, 2006; Feng & Hirst, 2012) tentent d'enrichir les modèles d'apprentissage avec des relations lexicales fines, du type de celles mentionnées plus haut, mais se heurtent à la faible couverture des ressources existantes.

Lister explicitement toutes les associations possibles de deux verbes dans cette perspective semble difficilement réalisable manuellement, et nous présentons une approche automatique, inspirée du projet Verbocean (Chklovski & Pantel, 2004), pour constituer une base lexicale large associant des verbes avec un typage d'ordre sémantique. Si notre objectif final est d'aider à la prédiction de relations discursives (rhétoriques), nous pensons utile de constituer cette ressource lexicale comme un intermédiaire intéressant d'autres tâches de nature sémantique. L'essentiel de la méthode est de recenser les paires de verbes dans des clauses adjacentes dans un grand corpus. Ces clauses sont souvent liées par des adverbiaux ou plus généralement des connecteurs discursifs marquant une ou plusieures relations discursives ; ces marqueurs suggèrent une relation discursive qui est enregistrée en association avec la paire. L' idée est de récupérer des paires de verbes qui sont souvent marquées avec une relation discursive (ou temporelle) et d'en déduire que cette paire de verbes peut suggérer une telle relation même en l'absence de marqueur. Ceci suppose que la relation repose non seulement sur le connecteur, mais également sur la paire de verbes employée : nous faisons ainsi l'hypothèse de redondance partielle du marqueur. Cette hypothèse a été précédemment discutée dans la littérature, notamment par (Sporleder & Lascarides, 2008) et (Braud, 2011), qui tendent à la soutenir.

La suite de l'article est organisée de la façon suivante. Nous détaillons d'abord la base de connaissances que nous avons construite (section 2), puis nous présentons ensuite les méthodes utilisées pour isoler des paires de verbes susceptibles d'apporter des informations discursives ou temporelles (section 3). Une troisième section décrit nos méthodes d'évaluation (section 4) et une quatrième fait une comparaison avec d'autres approches (section 5).

2 Explorer les relations entre verbes en corpus

La base de connaissances de relations verbales [2] a été construite à partir du corpus frWaC, qui fait partie de l'ensemble de corpora WaCKy (Baroni *et al.*, 2009). Celui-ci a été collecté sur le Web dans le domaine `.fr`, et contient environ 1.6 milliards de mots. Ce corpus a d'abord été parsé syntaxiquement grâce à la chaîne d'analyse de texte BONSAI [3] : étiquettage morpho-syntaxique par l'outil MElt (Denis & Sagot, 2012), puis analyse syntaxique en dépendances via une adaptation française du MaltParser (Nivre *et al.*, 2007). Le format de sortie est de type CONLL.

L'objectif est de rechercher des paires de verbes liés par une relation marquée explicitement par un connecteur dans le corpus. Les relations considérées sont des relations typiques en analyse discursive, éventuellement regroupée en groupes cohérents. Le corpus anglais du Penn Discourse TreeBank, le PDTB, (Prasad *et al.*, 2008) définit ainsi une hiérarchie d'une trentaine de relations comportant un niveau supérieur de quatre groupes : un groupe de relations causales (*contingency*), un groupe de relations temporelles, un groupe de "comparaison" (essentiellement des liens contrastifs), et un groupe d"'expansion" (essentiellement des types d'élaboration ou de continuation de discours). Afin de repérer les relations discursives marquées explicitement, on dispose en français du lexique LEXCONN (Roze *et al.*, 2012) [4], construit manuellement. Celui-ci rassemble 358 connecteurs du discours et comprend leurs catégories syntaxiques et les relations discursives associées, relations proches de celles de la SDRT (Asher & Lascarides, 2003). Certains connecteurs ont un usage ambigu, ils peuvent être associés à plusieurs relations. Dans un premier temps, dans un but de simplification, nous

1. Le corpus anglais PDTB par exemple comporte 52% de relations non marquées (Prasad *et al.*, 2008).
2. Disponible sous forme de base de données SQLite à `https://dl.dropboxusercontent.com/u/78938139/v2r_db`
3. Disponible à `http://alpage.inria.fr/statgram/frdep/fr_stat_dep_parsing.html`, cf aussi (Candito *et al.*, 2010)
4. Disponible librement : `https://gforge.inria.fr/frs/download.php/31052/lexconn.tar.gz`.

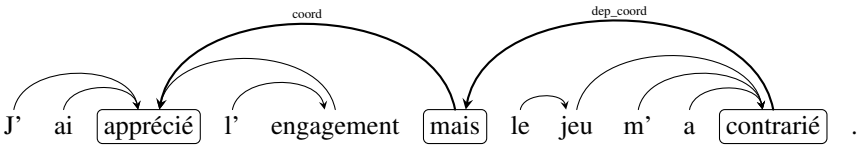
avons choisi de ne conserver que les connecteurs non ambigus, au nombre de 263. Par la suite, une désambiguisation pourra être opérée afin de permettre la prise en compte de tous les connecteurs. LEXCONN distingue une vingtaine de relations, et comme pour le PDTB, nous avons constitué des regroupements significatifs [5] : les relations d'explication (*parce que*) et de résultat (*ainsi*) forment le groupe causal, les relations d'organisation temporelle (*puis, après que*) ont été regroupées en un groupe de relations de narration. Les autres relations considérées sont des relations structurelles de contraste (*mais*), continuation (*et, encore*), arrière-plan (*alors que*), localisation temporelle (*quand, pendant que*), détachement (*de toutes façons*), élaboration (*en particulier*), alternation (*ou*), commentaire (*au fait*), reformulation (*du moins*), évidence (*effectivement*).

Un parcours du corpus parsé syntaxiquement est donc réalisé à la recherche de ces relations. Lorsqu'un connecteur est rencontré (après vérification de sa catégorie syntaxique), si celui-ci se trouve suffisamment proche de la racine de la phrase, une relation interphrastique est recherchée. Le premier verbe de la paire correspond alors au dernier verbe de la phrase précédente pour le cas des connecteurs de narration, ou au verbe principal de celle-ci pour toutes les autres relations. Le second verbe de la paire est recherché dans une fenêtre de deux liens de dépendances après le connecteur. Si le connecteur n'est pas suffisamment proche de la racine, une relation intraphrastique est recherchée. Pour cela, les deux verbes de la paire sont recherchés au sein de la même phrase, de part et d'autre du connecteur dans une fenêtre de deux liens de dépendances.

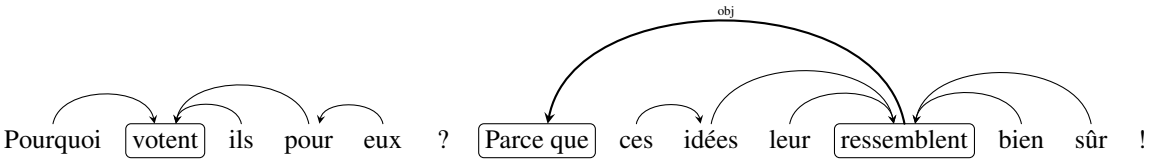
Dans le cas où deux lemmes verbaux sont effectivement identifiés, le contexte est examiné afin de mieux caractériser leur usage et d'affiner nos résultats. D'abord, si un verbe détecté est un modal ou un verbe support, la focalisation est reportée sur le verbe supporté par celui-ci, s'il existe, tout en mémorisant la présence du verbe support (qui ne constitue pas une distinction). La présence ou absence de négation et de particule réflexive constituent des critères de distinction entre les verbes (*comprendre* et *ne pas comprendre*, *agir* et *s'agir* sont des entrées distinctes). Par ailleurs, afin de distinguer les différents sens des verbes, deux types de traitement sont effectués. Une recherche d'un usage idiomatique de préposition est réalisée grâce à la ressource Dicovalence (Van Den Eynde & Mertens, 2010), qui répertorie les cadres de valence de plus de 3700 verbes simples du français (exemple : *tenir de* et *tenir à*). De plus la ressource Lefff (Lexique des Formes Fléchies du Français) (Sagot, 2010) permet de repérer les locutions verbales (exemples : *prendre garde*, *faire référence*). D'autres informations sont également mémorisées mais ne sont pas distinctives : temps du verbe, voix passive ou active.

Les deux exemples suivants illustrent cette méthode d'extraction (le schéma de dépendance est présenté, avec les liens utilisés pour l'extraction représentés en gras).

Exemple de relation intraphrastique :



Exemple de relation interphrastique :



Une fois la liste des paires associées à un connecteur obtenue, une agrégation de ces résultats est effectuée afin de les regrouper en types de triplets distincts (verbe 1, verbe 2, relation). Étant donné que seuls les connecteurs non ambigus ont été conservés, l'obtention de la relation associée est directe. À chaque triplet sont associés le nombre d'occurrences intraphrastiques, interphrastiques, et le nombre total d'occurrences. Les autres données récoltées mentionnées plus haut (temps, verbe support...) sont également conservées dans une table annexe.

Par cette méthode, plus de 2 millions d'occurrences de triplets, dont près de 95% intraphrastiques [6], ont été obtenues

5. Pour illustrer chaque relation nous donnons des exemples de marqueurs explicites entre parenthèses, potentiellement ambigu.

6. La faible proportion d'occurrences interphrastiques provient du schéma prudent choisi pour le repérage de ces occurrences, considérant uniquement les connecteurs présents en début de la seconde phrase. Nous avons en effet jugé le risque de produire du bruit en élargissant les schémas possibles trop important.

dans le corpus. Ces occurrences ont été regroupées en plus d'1 million de types de triplets distincts dans la base de connaissances. Parmi ces triplets, 4,5% présentent 5 occurrences ou plus.

Relation	Distribution
contraste	50,104%
cause	33,108%
continuation	8,243%
narration	6,362%
arrière-plan	1,853%
localisation temporelle	0.177%
détachement	0.149%
élaboration	0.002%
alternation	0.002%

TABLE 1 – La distribution des relations dans la base ; les relations commentaire, reformulation et évidence ont une fréquence d'apparition presque nulle.

La table 1 résume la distribution des triplets par relation. Nous pouvons noter que les relations de contraste et de cause sont largement majoritaires dans la base. Ceci ne signifie pas forcément que ce sont les plus présentes dans le corpus, mais plutôt qu'elles sont le plus fréquemment marquées par les connecteurs considérés. En prenant comme point de comparaison la ressource Annodis (Afantenos *et al.*, 2012), un corpus annoté manuellement en relations de discours, nous pouvons ainsi remarquer que les relations de continuation et d'élaboration sont bien plus fréquentes que dans nos annotations automatiques. Ceci signifie que ces relations reposent probablement moins sur la présence d'une marque explicite telle qu'un connecteur.

3 Mesurer l'association sémantique des paires de verbes

La section précédente présentait les données collectées sur les paires de verbes reliées par les marqueurs choisis, nous présentons maintenant les mesures testées pour classer la force d'association des paires de verbes. Nous nous sommes inspirés des mesures classiques d'association lexicale, issues de l'étude des cooccurrences, en les adaptant au contexte, et avons ajouté certaines mesures utilisées sur des relations spécifiques, comme les mesures de Do *et al.* (2011) pour détecter les liens de causalité entre verbes.

Les mesures d'association lexicale utilisées dans la recherche de cooccurrences servent à repérer des associations significatives, une fois que l'on tient compte de la fréquence des items reliés. Elles sont un composant essentiel des approches distributionnels de la sémantique et dans la construction d'espaces vectoriels de mots. Nous avons retenu une mesure simple, la PMI (*pointwise mutual information*) et ses variantes, *locale*, *normalisée*, *pondérée*, sensées atténuer les biais de la mesure originale (Evert, 2005). Le principe de la PMI est d'estimer si l'apparition simultanée de deux items est supérieure à la probabilité d'apparition *a priori* des deux items indépendamment. Nous avons appliqué cette mesure à des triplets constitués d'une paire de verbes avec une relation sémantique/discursive, parce que ce qui nous intéresse est de voir si la probabilité des deux items *avec* une relation sémantique particulière est supérieure à la probabilité d'apparition *a priori* des trois items indépendamment. Pour toutes nos mesures, nous considérons en fait l'événement consistant en l'apparition de deux items lexicaux dans une certaine relation indiquée par un marqueur explicite. Ceci est semblable aux approches syntaxiques en sémantique distributionnelle, qui pondèrent les associations d'items lexicaux dans une certaine relation syntaxique (comme nom-sujet-verbe, ou verbe-objet-nom).

$$PMI = \log(\frac{P(V_1, V_2, R)}{P(V_1) \times P(V_2) \times P(R)})$$

En cas de cooccurrence complète des trois items, nous avons :

$P(V_1) = P(V_2) = P(R) = P(V_1, V_2, R)$, et $PMI = -2\log(P(V_1, V_2, R))$.

Ainsi, la PMI normalisée est définie comme suit :

$$PMI\_normalisée = \frac{PMI}{-2\log(P(V_1, V_2, R))}$$

Notons ainsi que cette mesure est comprise entre -1 et 1, approchant -1 lorsque les items n'apparaissent jamais ensemble, prenant la valeur 0 en cas d'indépendance, et la valeur 1 en cas de cooccurrence complète.

La PMI pondérée proposée par Lin & Pantel (2002) est censée pallier le biais de la PMI pour les triplets peu fréquents :

$$PMI\_pondérée = discount \times PMI$$

$$discount = \frac{P(V_1,V_2,R)}{P(V_1,V_2,R)+1} \times \frac{\min[\sum_i(P(V_i,V_2,R)),\sum_i(P(V_1,V_i,R)),\sum_i(P(V_1,V_2,R_i)]}{\min[\sum_i(P(V_i,V_2,R)),\sum_i(P(V_1,V_i,R)),\sum_i(P(V_1,V_2,R_i))]+1}$$

La PMI locale quant à elle permet de prendre en compte la fréquence absolue d'occurrence du triplet :

$$PMI\_locale = F(V_1,V_2,R) \times \log(\frac{(V_1,V_2,R)}{P(V_1)\times P(V_2)\times P(R)}) = F(V_1,V_2,R) \times PMI$$

Nous nous sommes également inspirés d'une mesure de (Mirroshandel *et al.*, 2013), initialement définie pour mesurer la précision de cadres de sous-catégorisation, pour définir la mesure de spécificité :

$$spécificité = \frac{1}{3} \times (\frac{P(V_1,V_2,R)}{\sum_i P(V_1,V_i,R)} + \frac{P(V_1,V_2,R)}{\sum_i P(V_i,V_2,R)} + \frac{P(V_1,V_2,R)}{\sum_i P(V_1,V_2,R_i)})$$

Do *et al.* (2011) donnent une mesure complexe pour l'apport de deux prédicats qui supportent une relation causale, dont nous nous sommes inspirés dans la mesure suivante :

$$U_{do}(V_1,V_2,R) = PMI(V_1,V_2,R) \times max\{u_{V_1},u_{V_2},u_R\}$$

où : $u_{V_1} = \frac{P(V_1,V_2,R)}{\max_i(P(V_i,V_2,R))-P(V_1,V_2,R)+\varepsilon}$, $u_{V_2} = \frac{P(V_1,V_2,R)}{\max_i(P(V_1,V_i,R))-P(V_1,V_2,R)+\varepsilon}$

et $u_R = \frac{P(V_1,V_2,R)}{\max_i(P(V_1,V_2,R_i))-P(V_1,V_2,R)+\varepsilon}$.

Notons que la mesure initiale de (Do *et al.*, 2011) est également fonction de l'IDF (inverse document frequency), qui mesure la fréquence interdocument des deux verbes, et de la distance entre les deux instances. Ces deux derniers facteurs ne sont pas applicables à nos triplets, et ont donc été ignorés.

Nous avons également défini une mesure permettant d'évaluer l'apport de chaque composant du triplet à son informativité, similaire à la spécificité décrite ci-dessus.

$$W_{combinée}(V_1,V_2,R) = \frac{1}{3}(w_{V_1}+w_{V_2}+w_R)$$

Avec : $w_{V_1} = \frac{P(V_1,V_2,R)}{\max_i(P(V_i,V_2,R))}$, $w_{V_2} = \frac{P(V_1,V_2,R)}{\max_i(P(V_1,V_i,R))}$, et $w_R = \frac{P(V_1,V_2,R)}{\max_i(P(V_1,V_2,R_i))}$.

4 Évaluation des relations

Pour évaluer l'intérêt des paires de verbes extraites, nous avons procédé à plusieurs évaluations qui se veulent complémentaires. D'abord nous voulons une évaluation intrinsèque du lien entre les verbes, dans la perspective de valider la base comme une ressource sémantique, qui peut servir à des tâches différentes. Celle-ci est présentée ci-dessous (section 4.1). Nous présentons ensuite un début de validation extrinsèque, en étudiant l'impact potentiel de la ressource sur une tâche spécifique, à savoir la prédiction de relations discursives en l'absence de marque explicite (section 4.2).

4.1 Evaluation intrinsèque

Pour évaluer intrinsèquement les liens extraits, nous avons dans un premier temps étudié la possibilité d'attribuer fiablement un lien sémantique à une paire de verbes de façon "inhérente", c'est-à-dire hors de tout contexte Par exemple pour la

cause, est-il possible de juger qu'il y a une causalité "typique" entre les verbes *pousser* et *tomber*, dans des scénarios où ils partagent des arguments (sujet, objet, ...), ces scénarios étant laissés à l'appréciation du juge (section 4.1.1). Dans un deuxième temps, nous avons sélectionné quelques paires de verbes et une centaine de contextes dans lesquels ces paires apparaissent ensemble dans le corpus d'origine, pour juger du lien sémantique en contexte (section 4.1.2). Dans les deux cas, nous avons restreint l'étude à trois groupes de relations, causales, contrastives et narratives. Ce sont les plus couramment marquées dans le corpus, et elles constituent des cas assez différent de lien, en ayant une composante sémantique qui paraît significative (à l'inverse de la relation, très marquée également, de continuation).

4.1.1 Evaluation hors contexte des paires

Pour le jugement sur des liens hors contexte, nous avons suivi le protocole suivant : un des auteurs a choisi 100 paires de verbes avec des proportions similaires de paires présentant des bons et mauvais scores pour la relation choisie et selon les mesures choisies. Ensuite, les trois autres auteurs ont dû juger pour chacune des 300 paires si elle pouvait ou non être reliée avec la relation considérée, sans connaître l'origine des paires. La table résume les accords inter-annotateurs, estimés avec le kappa de Cohen (Carletta, 1996). Il est apparu assez vite que la tâche était très difficile voire infaisable pour la causalité, difficile pour la narration, et moyennement difficile pour le contraste, si l'on juge classiquement qu'un kappa autour de 0.6 est acceptable, surtout pour un jugement de nature sémantique.

Nous avons donc décidé d'ignorer ces jugements pour la cause et la narration, et avons gardé les jugements sur le contraste, après adjudication entre les trois annotateurs. Pour évaluer les mesures d'association choisies, nous avons testé statistiquement si elles discriminaient entre les deux groupes de paires de verbes (celles jugées positivement et négativement par les annotateurs). La table 3 résume ces tests, où l'on voit que toutes les mesures discriminent statistiquement les deux groupes, sauf les comptages bruts de cooccurrence.

Annotateurs	Cause	Contraste	Narration
1/2	0.16	0.55	0.43
1/3	0.22	0.57	0.46
2/3	0.13	0.56	0.37
kappa moyen	0.17	0.56	0.42

TABLE 2 – Accords inter-annotateurs pour annotations hors contexte : kappa par paires d'annotateurs et moyenne des kappas.

Mesure	valeur de p
spécificité	2.5e-11
U_do	2.9e-11
PMI_normalisée	1.28e-10
PMI_pondérée	1.96e-10
PMI	1.86e-10
W_combinée	4.93e-10
PMI_locale	4.95e-08
comptage d'occurrences inter-phrastiques	0.000904
comptage d'occurrences intra-phrastiques	0.0721
comptage d'occurrences brut	0.116

TABLE 3 – Tests de MannWhitney-U pour estimer la différence des mesures sur les groupes de paires de verbes contrastives ou non. Les mesures sont triées par valeur de p croissante, les valeurs dans la deuxième partie du tableau étant non significatives.

4.1.2 Evaluation en contexte

Pour une évaluation plus précise des liens sémantiques, nous avons aussi considéré des jugements d'association en contexte : en plus de faciliter le jugement, l'idée est aussi que le caractère typique de la relation entre deux verbes peut être

estimée par la proportion de contextes où ils apparaissent ensemble avec le lien que l'on considère. On peut ensuite observer si cette proportion est corrélée avec les mesures d'association présentées précédemment. L'écueil de cette méthode est son coût en annotation : si l'on veut évaluer un lien entre deux verbes spécifiques il faut déjà un certain nombre de contextes pour appuyer le jugement, et il faut répéter cette évaluation sur un nombre suffisant de paires de verbes. De plus, il faut avoir un échantillon de paires qui couvre suffisamment de valeurs différentes pour observer des corrélations significatives, alors qu'on ne peut préjuger des valeurs attribuées par l'annotation humaine. Nous avons finalement choisi 40 contextes exemples pour chacune des 15 paires de verbes sélectionnées (5 pour chaque relation : cause, narration, contraste) ; les paires sélectionnées le sont selon des scores échelonnés de PMI_normalisée, et encore une fois par l'un des auteurs indépendemment des trois autres, qui ont réalisé l'annotation séparément, avant de procéder à une adjudication des 600 contextes pour la référence. Pré-adjudication, l'accord brut sur les décision est de 78% en moyenne, pour un kappa moyen de 0,46 et un kappa maximum de 0,49. Ces valeurs semblent faibles, ce qui souligne la difficulté de la tâche. Le résultat de cette annotation est présenté table 4.

Verbe 1	Verbe 2	relation	association/humain
inviter	souhaiter	causale	12.8%
promettre	élire	causale	25.6%
aimer	trouver	causale	38.5%
bénéficier	créer	causale	51.3%
aider	gagner	causale	53.8%
proposer	refuser	contraste	59.0%
augmenter	diminuer	contraste	64.1%
tenter	échouer	contraste	64.1%
gagner	perdre	contraste	71.8%
autoriser	interdire	contraste	74.4%
parler	réfléchir	narration	42.5%
acheter	essayer	narration	70.0%
atteindre	traverser	narration	77.5%
commencer	finir	narration	80.0%
envoyer	transmettre	narration	82.5%

TABLE 4 – La liste des paires de verbes évaluées manuellement en contexte, avec la relation à juger et le ratio d'association résultant de l'adjudication humaine

Nous avons ensuite mesuré la corrélation entre cet indice d'association obtenu à partir de l'annotation humaine, et les mesures d'association présentées plus haut. Nous indiquons ici deux corrélations séparées : une première sur l'ensemble des données annotées, et une seconde sur le sous-ensemble des contextes ne comportant pas de marqueur de la relation considérée (les contextes implicites). Cette dernière mesure est importante pour quantifier l'apport effectif de la méthode suivie ici, et éviter la tautologie qui consisterait à trouver des liens marqués explicitement en corpus pour identifier les mêmes liens marqués dans des contextes particuliers. De fait, les contextes sans marques explicites sont les seuls à n'être pas intervenu dans le calcul des mesures d'association.

	Corrélation globale	Corrélation sur instances implicites
PMI_normalisée	0.749	0.806
spécificité	0.747	0.760
W_combinée	0.720	0.738
PMI_pondérée	0.716	0.761
PMI	0.709	0.756
PMI_locale	0.434	0.553
U_do	0.376	0.499
frequence brute	0.170	0.242

TABLE 5 – Les corrélations de Pearson pour les 15 paires considérées et pour les mesures présentées à la section 3, par ordre décroissant.

Ceci posé, on peut observer que les mesures d'information mutuelle sont bien corrélées, même la PMI simple, et que la mesure W_combinée que nous proposions section 3 est également utile. Nous avons également observé les résultats pour chaque relation séparément (résultats non détaillés ici), avec prudence à cause du faible nombre de points (5 par relation), et avons pu noter de grandes variations dans le comportement des mesures sur chaque relation. Notons ainsi que la mesure U_do, initialement formulée pour les relations causales, ne se généralise pas bien sur l'ensemble des relations ni sur les autres relations, par contre son bon fonctionnement a pu être vérifié pour les relations causales. Par ailleurs, la mesure de PMI_locale fonctionne très bien pour les relations de narration et de cause.

Ces résultats nous ont permis d'identifier les trois meilleures mesures : la PMI_normalisée, la spécificité et W_combinée. Nous avons observé que ces deux dernières assignent leur valeur maximale à de multiples paires. Nous avons alors imposé un ordre lexicographique en utilisant la PMI normalisée pour départager les meilleures paires. La table 6 présente ainsi les meilleures paires obtenues pour les relations de narration, continuation, cause et contraste.

Verbe 1	Verbe 2	Relation
abandonner	mener	arrière-plan
ne pas s'arrêter	rouler	narration
donner satisfaction sur	réélire	continuation
emporter	ne pas cesser	élaboration
emprunter	assurer	cause
ne pas manquer	prolonger	détachement
ratifier	trembler	arrière-plan
avoir honte	faire pitié	cause
avoir droit	cotiser pour	loc. temp.
ne pas représenter	stéréotyper	loc. temp.

TABLE 6 – Listes des 10 meilleures paires de la base selon notre ordre lexicographique.

4.2 Evaluation extrinsèque

L'objectif principal de notre travail est de constituer une ressource de relations sémantiques, dont la principale application visée est l'aide à la prédiction de relations rhétoriques. Afin d'évaluer la performance de notre base de triplets dans cette optique, nous avons pour perspective de l'utiliser comme traits additionnels dans un modèle de prédiction de relations. Au préalable, nous avons cherché à évaluer l'impact potentiel de notre ressource sur la tâche de prédiction, en étudiant sa couverture par rapport à un corpus en français annoté en relations de discours, le corpus Annodis (Afantenos *et al.*, 2012). Pour espérer améliorer les modèles existants, il faut en effet qu'une partie significative des relations à prédire implique des paires de verbes présentes dans la base constituée. Un indicateur fort du succès potentiel de la tâche est aussi la part de relations entre deux segments contenant deux verbes existant dans la base, et dont le lien majoritaire fait partie du groupe concerné par la relation, par exemple pour une relation d'explication, le fait que les deux verbes soient reliés par un lien causal. Ceci n'est intéressant qu'à la condition que l'instance en question ne présente pas déjà un marquage direct de la relation via un connecteur de discours, c'est-à-dire que la relation soit implicite. Par ailleurs, toujours dans le cas des liens implicites, il est intéressant d'avoir l'information que deux verbes sont dans un certain rapport sémantique, même si celui-ci ne correspond pas directement à un groupe lié à la relation recherchée : un lien causal potentiel entre deux événements peut informer au moins qu'une succession temporelle est pertinente pour considérer le lien entre deux unités de discours.

Pour mener cette étude, nous nous sommes reposés là encore sur la base de marqueurs Lexconn. En première approximation, nous avons considéré qu'une instance de relation entre deux segments de discours était explicite quand un connecteur de Lexconn était présent dans un des deux segments, et qu'un de ses sens recensés était celui de la relation présente dans le corpus. Cela peut au pire surestimer le nombre d'exemples explicites, et assure que les exemples implicites considérés le sont effectivement (à quelques erreurs de détection de marquage près). Pour simplifier nous n'avons considéré que les liens entre unités discursives simples, Annodis comportant également des relations entre ensembles de segments simples (segments dits "complexes").

La table 7 présente les résultats de couverture, pour les relations principales. Il faut noter qu'un petit nombre d'instances du corpus sont concernées (400 sur environ 2000 relations entre segments simples), les autres n'impliquant souvent qu'un

verbe, et certaines étant oubliées à cause d'erreur d'étiquetage ou de détection. Là encore ces chiffres sont à prendre comme des estimations conservatrices.

	global	narration	cause	contrast	elab.	cont.	AR	autres
paires dans annodis	407	72	65	40	97	93	24	16
paires annodis ∈ vpdb	66.8	65.3	67.7	72.5	69.1	60.2	79.2	62.5
triplets annodis∈ vpdb	33.7	34.7	49.2	65.0	0.0	19.4	8.3	0.0
triplets annodis marqués dans l'instance	20.4	29.2	21.5	62.5	1.0	5.4	12.5	0.0
paires annodis implicites (pas de connecteur/ou autre connecteur)	79.6	70.8	78.5	37.5	99.0	94.6	87.5	100.0
triplets annodis implicites ∈ vpdb (avec relation correcte)	24.6	23.6	40.0	27.5	0.0	18.3	8.3	0.0
paires annodis implicites ∈ vpdb (toutes relations)	53.1	48.6	50.8	27.5	68.0	57.0	70.8	62.5
verbes absents de vpdb	0.4	1.0	0.0	1.4	0.0	0.0	0.0	0.0

TABLE 7 – Couverture des paires de verbes dans la base (vpdb) par rapport aux instances du corpus Annodis impliquant deux verbes. Paire = paire de verbes dans des segments reliés par une relation rhétorique, Triplet=la paire de verbes associée à la relation rhétorique dans une instance du corpus. A part la première ligne, tous les chiffres sont en pourcentage. AR = arrière-plan, cont=continuation, elab=elaboration.

Nous avons listé plusieurs types d'information : la présence dans la base des paires de verbes des instances du corpus discursif, la présence des paires de verbes associés à la relation qui les lie dans l'instance d'Annodis considérée [7], c'est-à-dire la présence du triplet (verbe1,verbe2,relation) dans la base, et la restriction de ces statistiques aux contextes où il y a présence ou absence d'un marqueur explicite d'une relation.

Nous pouvons voir que presque tous les verbes présents dans le corpus discursif sont recensés dans la base dans au moins une paire (les deux exceptions sont un verbe non lemmatisé par le taggeur, et une locution verbale), mais que les paires recensées dans la base ne couvrent que partiellement les paires apparaissant dans Annodis. En effet entre 60 et 70% des instances associent des verbes présents dans la base (selon la relation), et un peu moins si l'on considère les instances implicites (environ 50% en moyenne), sauf la relation de contraste, fortement marquée dans Annodis. Il est très encourageant de noter qu'une forte proportion de contextes sans marquage contiennent des paires de verbes qui sont collectées dans un contexte marqué, même pour des relations peu marquées comme élaboration ou continuation, et qui plus est une bonne part de ces contextes (plus de la moitié) sont correctement associés à la bonne relation dans la base (éventuellement parmi d'autres). L'hypothèse de redondance partielle des connecteurs semble pouvoir être utile quand on considère le corpus dans son ensemble pour isoler des associations verbales pertinentes pour le discours. Tout ceci demande à l'évidence d'être poursuivi en intégrant ces informations à un étiquetteur de relations discursives, mais semble prometteur dans cette perspective.

5 Travaux reliés

Nous pouvons distinguer deux différents types de travaux reliés à notre approche. Pour le premier groupe, l'idée fondamentale est de pallier le manque de données annotées en utilisant une approche faiblement supervisée, exploitant la présence de marqueurs explicites dans un grand corpus non-annoté. Chaque paire d'unités discursives élémentaires est ainsi annotée automatiquement avec la relation discursive correspondant au marqueur (les marqueurs sont souvent filtrés par rapport à leurs usages non-discursifs). Ensuite, ces marqueurs sont éliminés du corpus afin d'empêcher les modèles de se baser sur cet indice, créant ainsi artificiellement des relations implicites. L'article pionnier de cette approche est celui de (Marcu & Echihabi, 2002). Les relations utilisées dans cet article correspondent à un niveau de granularité plus grossier par rapport aux relations typiquement utilisées en RST (Mann & Thompson, 1988), obtenant néanmoins des scores assez bas. La même approche a été aussi poursuivie par Sporleder & Lascarides (2008) obtenant des résultats à peine au-dessus du hasard comme l'ont montré Braud & Denis (2013). Ces derniers ont ainsi observé les performances relativement faibles de cette méthode de prédiction des relations implicites avec des données « artificielles » (relation explicite rendue

7. Chaque paire de verbes extraite d'annodis est présente une seule fois.

artificiellement implicite par suppression du marqueur) par rapport aux résultats obtenus avec des données « naturelles » (relation implicite annotée par un humain). Ils ont alors proposé une méthode consistant à combiner ces deux types de données soit au niveau du jeu de données soit directement au niveau de l'algorithme d'apprentissage, obtenant ainsi une amélioration significative sur le corpus ANNODIS. Notre approche est différente en ce qu'elle veut isoler explicitement les liens entre paires de verbes, pour élargir l'usage de cette information à d'autres tâches. Dans une prochaine étape nous chercherons cependant à mesurer l'apport de notre ressource pour cette tâche de prédiction de relations par rapport aux approches précédentes, permettant ainsi une évaluation extrinsèque.

Un deuxième groupe de travaux vise à identifier les relations discursives (implicites ou non) en se focalisant sur l'utilisation des relations lexicales fines comme un autre indice pendant la phase d'apprentissage. La plupart des travaux se concentrent principalement sur les relations lexicales entre deux verbes. Chklovski & Pantel (2004) par exemple, se sont appuyés sur des patrons spécifiques construits manuellement pour chaque relation sémantique parmi (*similarity, strength, antonymy, enablement* et *temporal happens-before*). Ensuite, le Web a servi de corpus afin d'estimer la PMI entre deux verbes et un patron (un calcul précis de ne peut pas être réalisable puisque la probabilité d'un verbe ou un patron sur tout le web ne peut être connue précisément). Un seuil (estimé manuellement) sur les valeurs de PMI a ainsi permis de déterminer les paires de verbes considérées comme liées par la relation indiquée par le patron. Dans le même esprit, Kozareva (2012) s'est basée sur une approche faiblement supervisée pour réaliser l'extraction de paires de verbes potentiellement impliqués dans une relation *cause-effet*. La méthode consiste à utiliser des patrons appliqués sur le Web pour extraire des paires et générer de nouvelles graines. Des travaux similaires ont été réalisés par Do *et al.* (2011), prenant cependant en compte non seulement les verbes mais aussi les noms dénotant un événement. Ils se sont concentrés sur les relations causales, utilisant les marqueurs discursifs comme indice. Selon leurs travaux, un événement est un prédicat avec un certains nombre d'arguments et donc l'association d'événements est la somme d'associations entre prédicats, entre prédicats et arguments et entre arguments. Toutes leurs mesures sont basées sur la PMI corrigée pour certains cas (paires trop fréquentes, distance textuelle entre les prédicats d'une paire, fréquence des prédicats). À l'aide du Gigaword comme corpus et d'une réimplémentation de (Lin *et al.*, 2014), ils ont alors extrait les relations discursives. Un système de programmation logique inductive est finalement utilisé, exploitant les interactions entre paires causales et relations discursives afin d'extraire les liens causaux. Ces travaux se concentrent donc sur des relations particulières, à l'exception de Chklovski & Pantel (2004), qui ne présentent pas d'évaluation systématique de leurs résultats.

Enfin, il faut mentionner les travaux qui se soucient directement de l'apprentissage des structures discursives mais qui enrichissent leur système en ajoutant de l'information lexicale. Feng & Hirst (2012) ont utilisé HILDA (Hernault *et al.*, 2010), y ajoutant d'avantage de traits. Une famille de traits représente la similarité lexicale fondée sur les distance dans les hiérarchies VERBNET et WORDNET. D'une façon similaire, Wellner *et al.* (2006) se sont focalisés sur les relations discursives intraphrastiques et ont ajouté de l'information lexicale dans des traits basés sur les mesures proposées par Lin (1998) et calculées sur le British National Corpus. Ces approches n'utilisent donc que des liens lexicaux de similarité, sans typage sémantique de ce lien, et l'impact de cette information seule semble limitée. Du point de vue de l'évaluation, notre méthode est assez proche de celle suivie dans les travaux sur la relation d'implication dans (Tremper & Frank, 2013), combinant évaluation hors et en contexte d'associations verbales. Les accords inter-annotateurs sont similaires aux nôtres (0.42-0.44 de Kappa), avec des choix légèrement différents : les annotateurs étaient censés discriminer le lien verbal entre les différents sous-cas possibles. Les paires de verbes étaient repérées par le système de Lin et Pantel. Ces auteurs présentent également un modèle de classification parmi les différents types de relations, en supposant donné le fait que deux verbes sont liés sémantiquement.

6 Conclusion

Nous avons présenté ici une base de connaissances comportant des triplets de paires de verbes associés avec une relation sémantique/discursive, extraits du corpus français frWaC par une méthode s'appuyant sur la présence d'un connecteur discursif reliant deux verbes. Nous avons détaillé plusieurs mesures visant à évaluer la pertinence des triplets et la force d'association entre la relation sémantique/discursive et la paire de verbes. Des évaluations par annotation manuelle nous ont permis de valider notre approche et de sélectionner les meilleures mesures. Nous avons également réalisé une étude de la couverture de la ressource par rapport aux triplets annotés manuellement du corpus Annodis. Ceci nous a permis de vérifier qu'un grand nombre de triplets implicites dans Annodis sont présents dans notre base de connaissances. Ces résultats positifs encouragent la poursuite de nos travaux dans la perspective d'utiliser cette ressource pour améliorer les méthodes d'analyse du discours, mais aussi pour des tâches de nature sémantique.

Références

AFANTENOS S., ASHER N., BENAMARA F., BRAS M., FABRE C., HO-DAC M., DRAOULEC A. L., MULLER P., PERY-WOODLEY M.-P., PREVOT L., REBEYROLLES J., TANGUY L., VERGEZ-COURET M. & VIEU L. (2012). An empirical resource for discovering cognitive principles of discourse organisation : the ANNODIS corpus. In N. CALZOLARI, K. CHOUKRI, T. DECLERCK, M. U. DOĞAN, B. MAEGAARD, J. MARIANI, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey : European Language Resources Association (ELRA).

ASHER N. & LASCARIDES A. (2003). *Logics of Conversation*. Studies in Natural Language Processing. Cambridge, UK : Cambridge University Press.

BAKER C. F., FILLMORE C. J. & LOWE J. B. (1998). The Berkeley FrameNet Project. In *Proceedings of the COLING-ACL*, Montreal, Canada.

BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, **43**(3), 209–226.

BRAUD C. (2011). Identification automatique des relations rhétoriques en français à partir de corpus annotés et de corpus bruts. Master's thesis, Université Paris Diderot.

BRAUD C. & DENIS P. (2013). Identification automatique des relations discursives "implicites" à partir de données annotées et de corpus bruts. In *TALN - 20ème conférence du Traitement Automatique du Langage Naturel 2013*, volume 1, p. 104–117, Sables d'Olonne, France.

CANDITO M., CRABBÉ B. & DENIS P. (2010). Statistical french dependency parsing : Treebank conversion and first results. In *LREC*, Valletta, Malta.

CARLETTA J. (1996). Assessing agreement on classification tasks : the kappa statistic. *Computational linguistics*, **22**(2), 249–254.

CHAMBERS N. & JURAFSKY D. (2008). Unsupervised Learning of Narrative Event Chains. In *Proceedings of ACL-08 : HLT*, p. 789–797, Columbus, Ohio.

CHKLOVSKI T. & PANTEL P. (2004). Verbocean : Mining the web for fine-grained semantic verb relations. In D. LIN & D. WU, Eds., *Proceedings of EMNLP 2004*, p. 33–40, Barcelona, Spain : Association for Computational Linguistics.

DENIS P. & SAGOT B. (2012). Coupling an annotated corpus and a lexicon for state-of-the-art pos tagging. *Language Resources and Evaluation*, (46), 721–736.

DO Q., CHAN Y. S. & ROTH D. (2011). Minimally supervised event causality identification. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, p. 294–303, Edinburgh, Scotland, UK. : Association for Computational Linguistics.

EVERT S. (2005). *The statistics of word cooccurrences*. PhD thesis, Stuttgart University.

FELBAUM C. (1998). *Wordnet, an Electronic Lexical Database for English*. Cambridge : MIT Press.

FENG V. W. & HIRST G. (2012). Text-level discourse parsing with rich linguistic features. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 60–68, Jeju Island, Korea : Association for Computational Linguistics.

GREFENSTETTE G. (1994). *Explorations in automatic thesaurus discovery*. Springer.

HASHIMOTO C., TORISAWA K., KURODA K., DE SAEGER S., MURATA M. & KAZAMA J. (2009). Large-scale verb entailment acquisition from the Web. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, p. 1172–1181, Singapore : Association for Computational Linguistics.

HERNAULT H., PRENDINGER H., DUVERLE D. A. & ISHIZUKA M. (2010). HILDA : A Discourse Parser Using Support Vector Machine Classification. *Dialogue and Discourse*, **1**(3), 1–33.

KOZAREVA Z. (2012). Cause-effect relation learning. In *Workshop Proceedings of TextGraphs-7 : Graph-based Methods for Natural Language Processing*, p. 39–43, Jeju, Republic of Korea : Association for Computational Linguistics.

LIN D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 36th ACL and 17th COLING joint conference*, volume 2, p. 768–774, Montreal.

LIN D. & PANTEL P. (2002). Concept discovery from text. In *Proceedings of Coling 2002*, p. 1–7 : Association for Computational Linguistics.

LIN Z., NG H. T. & KAN M.-Y. (2014). A PDTB-styled end-to-end discourse parser. *Natural Language Engineering*, **20**(2), 151–184.

LIU M., LI W., WU M. & LU Q. (2007). Extractive summarization based on event term clustering. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, p. 185–188, Prague, Czech Republic : Association for Computational Linguistics.

MANN W. C. & THOMPSON S. A. (1988). Rhetorical Structure Theory : Towards a Functional Theory of Text Organization. *Text*, **8**(3), 243–281.

MARCU D. & ECHIHABI A. (2002). An Unsupervised Approach to Recognizing Discourse Relations. In *Proceedings of ACL*, p. 368–375.

MIRROSHANDEL S. A., NASR A. & SAGOT B. (2013). Enforcing subcategorization constraints in a parser using sub-parses recombining. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 239–247, Atlanta, Georgia : Association for Computational Linguistics.

NIVRE J., HALL J., NILSSON J., CHANEV A., ERYIGIT G., KÜBLER S., MARINOV S. & MARSI E. (2007). Maltparser : A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, **13**(2), 95–135.

PRASAD R., DINESH N., LEE A., MILTSAKAKI E., ROBALDO L., JOSHI A. & WEBBER B. L. (2008). The Penn Discourse TreeBank 2.0. In *Proceedings of LREC 2008*.

ROZE C., DANLOS L. & MULLER P. (2012). Lexconn : A french lexicon of discourse connectives. *Discours*, (10).

SAGOT B. (2010). The lefff, a freely available and large-coverage morphological and syntactic lexicon for french. In *7th international conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta.

SPORLEDER C. & LASCARIDES A. (2008). Using Automatically Labelled Examples to Classify Rhetorical Relations : An Assessment. *Natural Language Engineering*, **14**(3), 369–416.

TREMPER G. & FRANK A. (2013). A discriminative analysis of fine-grained semantic relations including presupposition : Annotation and classification. *Dialogue & Discourse*, **4**(2), 282–322.

UZZAMAN N., LLORENS H., DERCZYNSKI L., ALLEN J., VERHAGEN M. & PUSTEJOVSKY J. (2013). Semeval-2013 task 1 : Tempeval-3 : Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2 : Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, p. 1–9, Atlanta, Georgia, USA : Association for Computational Linguistics.

VAN DEN EYNDE K. & MERTENS P. (2010). Le dictionnaire de valence : Dicovalence. http ://bach.arts.kuleuven.be/dicovalence/.

WELLNER B., PUSTEJOVSKY J., HAVASI C., RUMSHISKY A. & SAURÍ R. (2006). Classification of discourse coherence relations : an exploratory study using multiple knowledge sources. In *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, SigDIAL '06, p. 117–125, Stroudsburg, PA, USA : Association for Computational Linguistics.

Modèles de langue neuronaux: une comparaison de plusieurs stratégies d'apprentissage

Quoc-Khanh Do[1, 2] Alexandre Allauzen[1, 2] François Yvon[1]
(1) LIMSI/CNRS, rue John von Neumann, Campus Universitaire Orsay 91 403 Orsay
(2) Université Paris Sud, 91 403 Orsay
prenom.nom@limsi.fr

Résumé. Alors que l'importance des modèles neuronaux dans le domaine du traitement automatique des langues ne cesse de croître, les difficultés de leur apprentissage continue de freiner leur diffusion au sein de la communauté. Cet article étudie plusieurs stratégies, dont deux sont originales, pour estimer des modèles de langue neuronaux, en se focalisant sur l'ajustement du pas d'apprentissage. Les résultats expérimentaux montrent, d'une part, l'importance que revêt la conception de cette stratégie. D'autre part, le choix d'une stratégie appropriée permet d'apprendre efficacement des modèles de langue donnant lieu à des résultats à l'état de l'art en traduction automatique, avec un temps de calcul réduit et une faible influence des hyper-paramètres.

Abstract. If neural networks play an increasingly important role in natural language processing, training issues still hinder their dissemination in the community. This paper studies different learning strategies for neural language models (including two new strategies), focusing on the adaptation of the learning rate. Experimental results show the impact of the design of such strategy. Moreover, provided the choice of an appropriate training regime, it is possible to efficiently learn language models that achieves state of the art results in machine translation with a lower training time and a reduced impact of hyper-parameters.

Mots-clés : Réseaux de neurones, modèles de langue n-gramme, traduction automatique statistique.

Keywords: Neural networks, n-gram language models, statistical machine translation.

1 Introduction

Après quelques années de relatif désintérêt, les réseaux de neurones artificiels ont retrouvé une place centrale dans le paysage de l'apprentissage automatique et ont récemment permis des avancées significatives pour de nombreux domaines applicatifs. Pour ce qui concerne le traitement automatique des langues (TAL), les applications sont également nombreuses et variées. Historiquement, les modèles de langue neuronaux ont été une des premières réalisations marquantes, avec des applications en reconnaissance automatique de la parole (RAP), depuis les travaux pionniers de (Nakamura *et al.*, 1990) jusqu'aux développements ultérieurs de (Bengio *et al.*, 2003; Schwenk, 2007; Mnih & Hinton, 2007; Le *et al.*, 2011; Mikolov *et al.*, 2011). Les modèles neuronaux ont été également appliqués à d'autres tâches complexes de modélisation, comme par exemple l'analyse syntaxique (Socher *et al.*, 2013), l'estimation de similarité sémantique (Huang *et al.*, 2012), les modèles d'alignement de mots (Yang *et al.*, 2013) ou encore en traduction automatique statistique (TAS) (Le *et al.*, 2012a; Kalchbrenner & Blunsom, 2013).

Les modèles de langue continue de jouer un rôle prépondérant dans de nombreuses applications du TAL, depuis la reconnaissance vocale jusqu'à la correction d'orthographe ou encore la traduction automatique. Les approches usuelles reposent sur des modèles n-grammes discrets, estimés avec des méthodes de lissage (Chen & Goodman, 1998). Pour ces modèles, l'occurrence d'un mot dans son contexte est considérée comme la réalisation d'une variable aléatoire discrète, dont l'espace de réalisation est le vocabulaire tout entier et au sein duquel il n'existe aucune relation entre les mots. Par exemple, aucune information statistique n'est partagée entre formes morphologiquement ou sémantiquement apparentées. Le caractère très inégal des distributions d'occurrences dans les textes implique que les modèles résultants sont souvent estimés à partir de petits nombres d'occurrences et possèdent une faible capacité de généralisation. Par ailleurs, le nombre de paramètres d'un modèle n-gramme augmentant de manière exponentielle avec son ordre, il est illusoire d'espérer

résoudre ce problème en augmentant la quantité des ressources textuelles disponibles [1]. Par opposition, les modèles de langue neuronaux (Bengio *et al.*, 2003) se caractérisent par une méthode d'estimation alternative des qui se fonde sur une représentation *continue* [2], puisque chaque mot du vocabulaire est représenté comme un point dans un espace métrique. La probabilité n-gramme d'un mot est alors une fonction des représentations continues des mots qui composent son contexte. Ces représentations, ainsi que les paramètres de la fonction d'estimation, sont apprises conjointement par un réseau de neurones multi-couche ; une stratégie d'estimation qui permet que les mots partageant des similarités distributionnelles auront des représentations proches. Ainsi, ce type de modèle introduit la notion de similarité entre mots et son exploitation permet une meilleure exploitation des données textuelles. L'intégration de ce type de modèles a permis des améliorations systématiques et significatives des performances en RAP et en TAS (Schwenk, 2007; Le *et al.*, 2011, 2012a). De plus, les représentations continues peuvent servir à de nombreuses tâches, comme par exemple l'étiquetage en parties du discours et en rôle sémantique (voir (Turian *et al.*, 2010; Collobert *et al.*, 2011) pour une vue d'ensemble).

Malgré toutes ces qualités, le coût computationel des modèles neuronaux, tant à l'apprentissage qu'à l'inférence, ainsi que la difficulté à mettre en œuvre une stratégie d'apprentissage efficace limitent la diffusion de ce type d'approche. Plusieurs solutions ont été proposées afin de réduire ce coût, comme l'utilisation de *short-list* impliquant les modèles neuronaux uniquement pour les mots les plus fréquents (Schwenk, 2007), la modification du critère d'optimisation (Mnih & Teh, 2012), ou encore l'usage d'une couche de sortie structurée (Mnih & Hinton, 2008) comme le modèle SOUL (Le *et al.*, 2011). Ces approches partagent néanmoins le même algorithme d'estimation, qui repose sur la maximisation de l'entropie croisée, réalisée par descente de gradient stochastique. Si son implantation est relativement aisée, cet algorithme nécessite de fixer au préalable un certain nombre d'hyper-paramètres, qui ont une grande influence sur la vitesse de convergence et sur les performances finales. Or, le choix des hyper-paramètres reste très empirique et une part importante du coût computationel consommé lors de l'apprentissage est dû à l'exploration de différents jeux d'hyper-paramètres afin d'en garder le plus approprié. Cela peut en rebuter certains, tant il est vrai que fixer ces hyper-paramètres est complexe et relève d'une expertise difficile à décrire et à transmettre. Parmi ces hyper-paramètres, le choix du pas d'apprentissage (*learning rate*) est crucial puisqu'il permet d'effectuer un compromis entre la vitesse de convergence et les performances (Schaul *et al.*, 2012) : même si l'optimisation ne converge pas vers un meilleur modèle, une stratégie appropriée et adaptative pour régler ce pas permet d'accélérer grandement la convergence, réduisant d'autant le coût computationel.

L'objectif de cet article est, dans un premier temps, de décrire de manière unifiée les différentes stratégie d'ajustement du pas d'apprentissage de la littérature, et d'en proposer deux nouvelles. Dans un second temps, grâce aux résultats expérimentaux, nous souhaitons montrer que certaines stratégies peuvent être à la fois peu dépendantes des choix initiaux pour les hyper-paramètres et en même temps donner lieu à des performances état de l'art, tant en terme de perplexité qu'en terme de score BLEU lorsque le modèle est utilisé au sein d'un système de TAS. Le reste de l'article est organisé de la manière suivante : la section 2 décrit les modèles de langue neuronaux qui seront utilisés dans cette étude ; puis les différentes stratégies d'ajustement du pas d'apprentissage sont passées en revue à la section 3 ; les résultats expérimentaux sont enfin présentés à la section 4.

2 Modèles de langue neuronaux

On s'intéresse aux modèles de langue neuronaux de type n-grammes [3] dont les probabilités sont estimées dans un espace continu. Introduits par (Bengio *et al.*, 2003), ces modèles se distinguent des modèles discrets par leur aptitude à prendre en compte un large contexte. Il est ainsi possible d'entraîner un modèle 10-gramme, puisqu'ici le nombre de paramètres ne croît que linéairement avec l'ordre du modèle. En revanche, une limitation des modèles décrits par (Bengio *et al.*, 2003; Schwenk, 2007), est que la taille du vocabulaire de prédiction est limitée afin de rendre le temps de calcul acceptable. Le modèle SOUL (*Structured OUtput Layer*), proposé par (Le *et al.*, 2011) permet de lever cette contrainte grâce à une représentation hiérarchique du vocabulaire de sortie. C'est ce modèle que nous étudions dans cet article.

2.1 Le modèle neuronal standard

Un modèle n-gramme calcule itérativement la probabilité que le mot w_i apparaisse dans un contexte determiné par les $n-1$ mots $\mathrm{w}_{i-\mathrm{n}+1}^{i-1}$ qui le précèdent, soit $P(\mathrm{w}_i|\mathrm{w}_{i-\mathrm{n}+1}^{i-1})$. Le modèle neuronal standard, tel qu'il est décrit dans (Bengio

1. Disposer de plus gros corpus reste naturellement un moyen sûr d'améliorer les performances (Brants *et al.*, 2007).
2. Les modèles neuronaux sont souvent qualifiés de modèles continus.
3. Par opposition aux modèles *récurrents* (Mikolov *et al.*, 2011), dont le contexte est a priori illimité. Une comparaison récente entre ces deux approches a montré qu'elles obtiennent des performances voisines, mais que le modèle n-gramme passe plus facilement à l'échelle (Le *et al.*, 2012b).

et al., 2003), réalise ce calcul de la manière suivante. Chaque mot du contexte en entrée du réseau est encodé par un vecteur **w** de dimension $|V|$, où $|V|$ est la taille du vocabulaire ; **w** contient une seule valeur non-nulle (typiquement égale à 1), sa position dans le vecteur permettant de repérer le mot parmi le vocabulaire (voir la figure 1) [4]. Les mots du contexte sont alors projetés dans un espace continu de dimension m par multiplication avec la matrice **R** de dimension $|V| \times$ m. Cette étape consiste en fait à sélectionner, dans la matrice **R**, le vecteur continu représentant chaque mot du contexte. Ainsi, un mot du vocabulaire est représenté par un vecteur dense à valeurs dans $\mathbb{R}^m$. En pratique, m $\ll |V|$ (500 au lieu de quelques centaines de milliers), ce qui explique la réduction considérable du nombre de paramètres du modèle. Dans un deuxième temps, ces $n-1$ représentations continues sont concaténées pour former $\mathbf{i}^{(1)}$, la première couche cachée du réseau. Un réseau multi-couche peut être vu comme un empilement de couches neuronales, chaque couche se définissant par un vecteur d'entrée $\mathbf{i}^{(l)}$, sa matrice de connexion $\mathbf{W}^{(l)}$, un vecteur de biais associé $\mathbf{b}^{(l)}$ et un vecteur de sortie $\mathbf{o}^{(l)}$ calculé par :

$$\mathbf{o}^{(l)} = \sigma\left(\mathbf{W}^{(l)}\mathbf{i}^{(l)} + \mathbf{b}^{(l)}\right), \tag{1}$$

où l désigne le numéro de la couche et σ est une fonction d'activation non-linéaire qui s'applique à chaque composante d'un vecteur. La couche de sortie **o** possède un neurone par mot à prédire (w_i) et la probabilité qui lui est associée est calculée grâce à une fonction d'activation de type *softmax* :

$$\mathbb{P}_{\boldsymbol{\theta}}(\mathrm{w}_i|\mathrm{w}_{i-\mathrm{n}+1}^{i-1}) = \exp(\mathbf{o}_{\mathrm{w}_i}) / \sum_{\mathrm{w}\in\mathbf{V}} \exp(\mathbf{o}_{\mathrm{w}}). \tag{2}$$

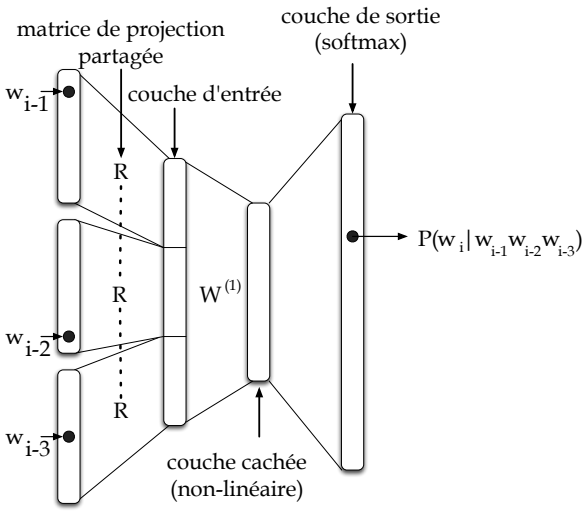


FIGURE 1: Architecture d'un modèle de langue neuronal standard pour $n = 4$

2.2 Le modèle SOUL

La comparaison des équations (1) et (2) aide à comprendre pourquoi la complexité d'un tel modèle se concentre dans la couche de sortie : le calcul d'une probabilité n-gramme nécessite d'abord une multiplication matricielle dont une des dimension est la taille du vocabulaire, puis la somme sur l'ensemble du vocabulaire pour normaliser la distribution n-gramme. Afin de réduire le temps de calcul, (Morin & Bengio, 2005; Mnih & Hinton, 2008) proposent de représenter le vocabulaire de sortie par un arbre binaire, représentant chaque mot par un chemin allant de la racine à une feuille. Dans cet arbre, chaque nœud interne représente une classe ou une sous-classe de mots. Le réseau de neurone peut alors servir à calculer itérativement la probabilité de ce chemin, plutôt que calculer directement celle du mot. Ces travaux ont été étendus par Le *et al.* (2011), qui proposent la structure SOUL (Figure 2). Cette structure se singularise par deux aspects : le modèle s'appuie sur un arbre d'arité quelconque et non simplement binaire ; une couche *softmax* est utilisée à chaque nœud de l'arbre au lieu d'une couche par niveau.

Compte-tenu des distributions très inégales des occurrences des mots, il reste souvent plus efficace de traiter à part les mots les plus fréquents. Ainsi, le modèle SOUL garde la notion de *short-list* regroupant les mots les plus fréquents du vocabulaire. Chacun de ces mots constitue une classe (singleton), comme représenté sur la partie droite de la figure 2.

4. Par la suite, les lettres en gras désignent un vecteur ou une matrice, selon qu'elles sont en minuscule ou majuscule.

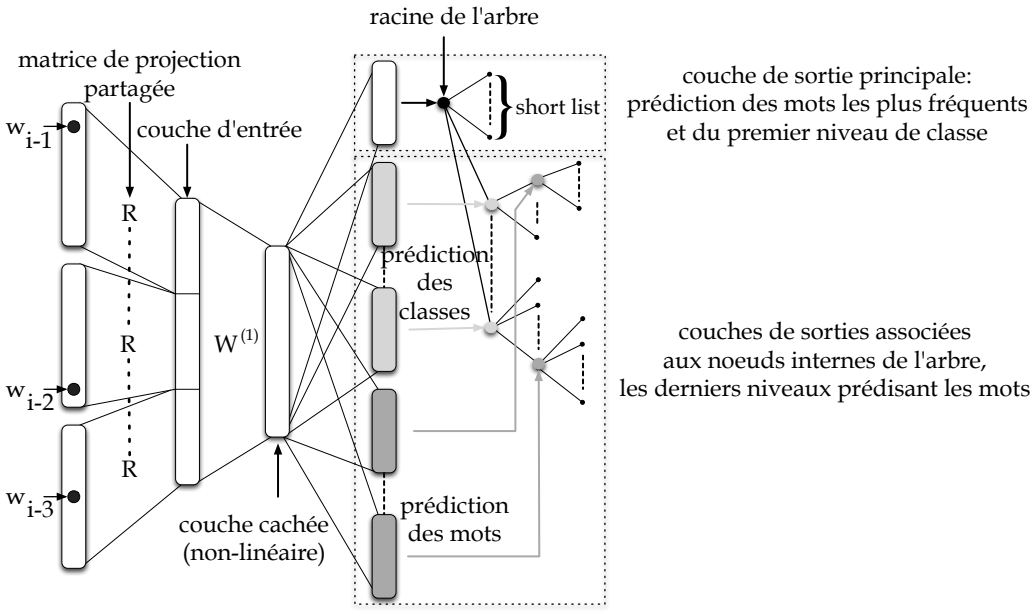


FIGURE 2: Architecture d'un modèle de langue SOUL pour $n = 4$

2.3 Apprentissage

L'apprentissage d'un modèle de langue neuronal se fonde sur la maximisation de la log-vraisemblance des paramètres ; l'objectif est alors de maximiser la fonction objectif définie ainsi :

$$\mathrm{L}(\boldsymbol{\theta}) = \sum_{\mathrm{w}_{i-\mathrm{n}+1}^{i} \in \mathbb{D}} \log \mathbb{P}_{\boldsymbol{\theta}}(\mathrm{w}_i | \mathrm{w}_{i-\mathrm{n}+1}^{i-1}) - \mathbb{R}(\boldsymbol{\theta}), \tag{3}$$

où $\boldsymbol{\theta}$ est le vecteur regroupant l'ensemble des paramètres du réseau neuronal : matrices de représentation et de connexion, ainsi que les biais. Le terme de régularisation (parfois nommé *weight decay*) $\mathbb{R}(\boldsymbol{\theta})$ correspond au carré de la norme euclidienne du vecteur de paramètres. Les termes de biais $\mathbf{b}^{(l)}$ de chaque couche l sont usuellement exclus de la régularisation. La méthode la plus utilisée pour maximiser (3) est la descente de gradient stochastique (Bottou, 2012). Le vecteur de paramètres $\boldsymbol{\theta}$ est mis à jour à l'instant $t+1$ de la manière suivante :

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} + \eta \nabla \mathrm{L}(\boldsymbol{\theta}^{(t)}), \tag{4}$$

où $\nabla \mathrm{L}(\boldsymbol{\theta}^{(t)})$ désigne le gradient de la fonction objectif pour la valeur des paramètres à l'instant t. Le pas d'apprentissage sur lequel nous focalisons notre étude est noté η. L'estimation du gradient et de la mise à jour peut se faire soit sur l'ensemble des données d'apprentissage (mode *batch*), soit sur quelques centaines d'exemples (mode *mini-batch*) ou bien pour chaque exemple (mode *online*). Par la suite nous considérons l'approche *mini-batch* qui offre un compromis entre garantie théorique et rapidité de convergence.

Les méthodes basées sur le Hessien demandent de calculer les dérivées secondes et ont été également envisagées pour l'apprentissage des réseaux neuronaux. Les algorithmes dit de Newton remplacent ainsi le pas d'apprentissage η dans l'équation (4) par l'inverse de la matrice Hessienne estimée à l'instant t :

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} + \left(\mathrm{H}^{(t)}\right)^{-1} \nabla \mathrm{L}(\boldsymbol{\theta}^{(t)}). \tag{5}$$

La portée de ce type d'approche pour les modèles neuronaux est limitée par un coût calculatoire exorbitant. En effet, ces méthodes requièrent l'estimation et le stockage de la matrice Hessienne qui est de dimension $|\boldsymbol{\theta}|^2$. À titre d'exemple, dans nos expériences, la matrice de projection $\mathbf{R}$ regroupe les représentations continues de dimension 500 pour chacun des $372K$ mots du vocabulaire et contient environ 1.8×10^8 paramètres. Par contre, les méthodes Quasi-Newton proposent une approximation du Hessien $\mathrm{H}^{(t)}$. Par exemple, une approximation diagonale de la matrice Hessienne équivaut à utiliser un pas d'apprentissage par paramètre. Ce type d'approche sera décrite dans la section suivante. Pour une vue d'ensemble de ces méthodes appliquées aux modèles neuronaux, le lecteur peut se référer à (LeCun *et al.*, 1998; Bengio, 2012).

L'apprentissage du modèle SOUL, s'il repose également sur la maximisation de la fonction de vraisemblance par des méthodes de gradient stochastique, implique en fait 3 étapes décrites dans (Le *et al.*, 2013) et que nous résumons ici :

1. apprentissage en quelques itérations [5] d'un modèle standard avec *short-list* : le vocabulaire de sortie est réduit aux quelques milliers de mots les plus fréquents. L'objectif est de disposer une première estimation des représentations continues des mots (**R**).
2. construction de l'arbre de partitionnement du vocabulaire. Pour cela, une version récursive de l'algorithme des K-moyennes est appliquée au vocabulaire, chaque mot étant représenté dans l'espace continu.
3. apprentissage du réseau global.

2.4 Représentations continues des mots et leur utilité

Afin de comprendre en quoi les représentations continues apprises (notées **R** à la section précédente) peuvent être utiles à différentes tâches du TAL, le tableau 1 décrit le voisinage de mots choisis au hasard, en français et en espagnol. Les représentations continues sont apprises par un modèle de langue SOUL utilisé lors des dernières campagnes d'évaluation WMT (voir pour plus de détails (Allauzen *et al.*, 2013)). Notons que les représentations sont apprises afin de modéliser le caractère distributionnel des mots puisqu'il s'agit d'un modèle n-gramme. Selon (Collobert *et al.*, 2011), la tâche d'apprentissage qui sert de supervision a un impact sur les représentations apprises et donc sur le type de similarités capturées. Une piste prometteuse de recherche pour apprendre ces représentations est de considérer différentes tâches, afin de pouvoir modéliser différentes caractéristiques linguistiques des mots.

mot	**voisins les plus proches**
évolution	décroissance - dynamisation - structuration - inflexion - atonie - fléchissement - infléchissement - hétérogénéité - réalignement
biotechnologie	nanotechnologie - photonique - bioinformatique - protéomique - nanotechnologies - microélectronique - géomatique - biotechnologies - microfinance
64	63 - 66 - 55 - 62 - 51 - 60 - 52 - 57 - 53
Renault	VW - Michelin - Nissan - Mercedes-Benz - Safran - Renault-Nissan - Faurecia - Volkswagen - Thalès
ouvre	ouvrait - ouvrent - ouvrira - ouvrir - ouvrirait - ouvraient - ouvrant - ouvriront - ouvrit
afiliados	agremiados - afiliadas - adheridos - afiliado - afiliada - adherentes - adscritas - empadronados - cotizantes
tristeza	angustia - desesperanza - desánimo - desilusión - amargura - asombro - estupor - resignación - incredulidad
debatiendo	discutiendo - examinando - debata - discutimos - debatimos - debatamos - debatiremos - debatió - debaten
Prado	Valme - Casal - Barranco - Atienza - Cabezas - Alcázar - Martirio - Eroski - Bermejales
Líbano	Argelia - Tíbet - Zimbabue - Jordania - Chechenia - Golán - Darfur - palestinas - OLP

TABLE 1: Exemples de voisinages dans l'espace de représentation continu pour des mots choisis aléatoirement, pour le français puis pour l'espagnol. Pour chaque mot considéré, le voisinage est défini ici par les 9 mots les plus proches.

3 Stratégies d'ajustement du pas d'apprentissage

Dans la plupart des cas, les modèles neuronaux utilisent un pas d'apprentissage dont la valeur initiale est fixée empiriquement, puis adaptée au cours de l'apprentissage via des stratégies simples. À notre connaissance, il n'existe pas de

5. Une itération correspond à un passage sur les données d'apprentissage. L'itération sert souvent d'unité de temps pour l'apprentissage des modèles neuronaux.

comparaisons de ces différentes stratégies qui permettrait d'évaluer leur impact sur les performances et sur le temps d'apprentissage. Pourtant, la littérature qui traite de la descente de gradient stochastique s'accorde sur le fait que le pas de gradient a un fort impact sur la vitesse de convergence (Robbins & Monro, 1951). De plus, une bonne stratégie d'ajustement permet de s'affranchir de l'influence de la valeur initiale sans altérer les performances du modèle (Schaul *et al.*, 2012). Ces stratégies d'ajustement peuvent se répartir en deux groupes : celles qui utilisent un pas d'apprentissage partagé par tous les paramètres, et d'autres qui ajustent un pas d'apprentissage par paramètre ou par groupe de paramètres.

3.1 Un pas d'apprentissage global

La stratégie la plus utilisée (**Power Scheduling**) impose la décroissance suivante : $\eta^{(t)} \sim t^{-1}$. Il a été montré que cette stratégie donnait la meilleure convergence asymptotique dans le cas d'un pas d'apprentissage global (Xu, 2011). De manière plus précise, l'instant t indique le nombre de mises à jour effectuées, et le pas d'apprentissage à l'instant t se calcule de la manière suivante : $\eta^{(t)} = \eta^{(0)}/(1 + \tau t)$, avec τ le taux de décroissance (*learning rate decay*). Dans (Bengio *et al.*, 2003), les auteurs fixent cet hyper-paramètre empiriquement à la valeur 10^{-8}. Le *et al.* (2012b) utilisent une variante de cette stratégie : $\eta^{(t)} = \eta^{(0)}/(1 + \tau \mathrm{N}_e^{(t)})$, avec $\mathrm{N}_e^{(t)}$ le nombre d'exemples d'apprentissage vu jusqu'à l'instant t.

La stratégie heuristique (**Down Scheduling**), qui est utilisée pour les modèles SOUL, est à la fois plus simple et plus efficace. Cette stratégie s'appuie sur la perplexité estimée sur un corpus de validation : lorsque la perplexité recommence à croître, le pas d'apprentissage est divisé par 2 après chaque itération. Une variante plus élaborée (**Adjust Scheduling**) et souvent nommée adaptative (e.g., (Ollivier, 2013)) est la suivante : après chaque itération, si la perplexité mesurée sur le corpus de validation augmente, les mises à jour de cette itération sont annulées et le pas d'apprentissage est divisé par 2 ; sinon le pas d'apprentissage est multiplié par 1,1. Ces deux stratégies permettent de réduire l'ensemble des hyper-paramètres au seul pas d'apprentissage initial. Enfin, la stratégie la plus simple (**Fix Scheduling**) consiste à choisir empiriquement une valeur du pas d'apprentissage qui restera fixe tout au long des itérations.

3.2 Ajustement du pas d'apprentissage pour chaque paramètre

Il est souvent recommandé d'utiliser différents pas d'apprentissage selon la partie du réseau impliquée (LeCun *et al.*, 1998; Bottou, 2012). Ce type d'approche peut être considérée comme un compromis entre la descente de gradient stochastique et les méthodes de Gauss-Newton ou Quasi-Newton. Cela revient en effet à approcher le Hessien par une matrice diagonale. Ainsi, LeCun *et al.* (1998) recommandent pour différentes tâches de classification l'usage des algorithmes de Gauss-Newton ou de Levenberg-Marquardt. Plus récemment, l'algorithme **AdaGrad** (Adaptive Gradient) (Duchi *et al.*, 2011) a montré une certaine efficacité dans plusieurs applications du TAL (Green *et al.*, 2013; Socher *et al.*, 2013). Dans sa forme originale, le Hessien est approché de la manière suivante :

$$\mathrm{H}^{(t)} = \left(\sum_{i=1}^{t} \nabla \mathrm{L}(\boldsymbol{\theta}^{(i-1)}) \nabla \mathrm{L}(\boldsymbol{\theta}^{(i-1)})^T \right)^{1/2}.$$

AdaGrad est souvent utilisé dans sa forme diagonale, car il est alors d'un point de vue calculatoire peu coûteux. Plus précisément, notons $\mathrm{g}_j^{(t-1)}$ le gradient de la fonction objectif par rapport au paramètre $\theta_j^{(t-1)}$ à l'instant $t-1$, le pas d'apprentissage à l'instant t associé au paramètre θ_j est calculé ainsi :

$$\eta_j^{(t)} = \frac{\eta^{(0)}}{\left(1 + \mathrm{G}_j^{(t)}\right)^{1/2}}, \tag{6}$$

$$\mathrm{G}_j^{(t)} = \sum_{i=1}^{t} \left(\mathrm{g}_j^{(i-1)}\right)^2, \tag{7}$$

où $\eta^{(0)}$ est la valeur initiale du pas d'apprentissage (commun à tous les paramètres) et $\mathrm{G}_j^{(t)}$ représente l'accumulation des gradients estimés par le passé ($\mathrm{g}_j^{(i-1)}, i = 1...t$).

3.3 Bloc-AdaGrad pour la structure SOUL

Afin d'adapter l'algorithme AdaGrad à la structure SOUL, nous en proposons une variante qui définit un pas d'apprentissage par groupe (ou bloc) de paramètres. Ce choix est motivé par l'objectif de réduction du temps de calcul. En effet, les opérations matricielles implémentées dans la bibliothèque BLAS jouent un rôle essentiel dans cette réduction. Nous choisissons les groupes de paramètres correspondant à ceux dont les gradients sont stockés dans une même matrice et sont calculés par une opération matricielle par BLAS ; tel que l'ajustement d'un pas d'apprentissage pour chacun de ces groupes n'ajoute qu'une complexité insignifiante comparé aux configurations où un seul pas d'apprentissage est utilisé pour l'ensemble de paramètres du réseau. L'ajustement d'un pas d'apprentissage pour chaque paramètre, comme proposé par AdaGrad, augmenterait considérablement le temps d'apprentissage. Dans cet algorithme, le terme η_j associé au paramètre θ_j dans l'équation (6) est remplacé par η_{b_j}, le pas d'apprentissage associé à tous les paramètres du bloc b_j :

$$\eta_{\mathrm{b}_j}^{(t)} = \frac{\eta^{(0)}}{\left(1 + \mathrm{G}_{\mathrm{b}_j}^{(t)}\right)^{1/2}} \tag{8}$$

$$\mathrm{G}_{\mathrm{b}_j}^{(t)} = \sum_{i=1}^{t} \left(\mathrm{g}_{\mathrm{b}_j}^{(i-1)}\right)^2 = \frac{1}{|\mathrm{b}_j|} \sum_{i=1}^{t} \sum_{k \in \mathrm{b}_j} \left(\mathrm{g}_k^{(i-1)}\right)^2 \tag{9}$$

Pour chaque couche cachée et pour chaque couche de sortie, deux blocs de paramètres sont définis : un pour la matrice de connexion et un pour le vecteur de biais. Pour la matrice de projection **R**, un bloc est défini pour chaque mot, soit pour chaque représentation continue. Cette séparation en blocs n'introduit pas de calcul supplémentaire, puisque les mises à jour sont également séparées lors de la descente stochastique de gradient.

4 Expériences

Afin d'analyser l'impact des différentes stratégies d'ajustement du pas d'apprentissage décrites à la section 3, une série d'expériences est menée sur la modélisation de l'espagnol avec, comme application, la tâche de traduction automatique de l'anglais vers l'espagnol, telle qu'elle est définie par la campagne d'évaluation WMT 2013 [6]. Différents critères sont utilisés pour l'évaluation : la perplexité du modèle de langue mesurée sur un corpus de validation, la vitesse de convergence en nombre d'itération, et la dépendance de ces critères aux choix des hyper-paramètres. Le modèle neuronal peut également être évalué de manière extrinsèque lorsqu'il est utilisé comme modèle additionnel en traduction automatique.

Nous utilisons ici un modèle neuronal 10-gramme utilisant le même vocabulaire (372K mots) que le modèle conventionnel discret. Le système de traduction utilisé est décrit dans (Allauzen *et al.*, 2013), les résultats obtenus le placent parmi les meilleurs systèmes pour cette tâche.

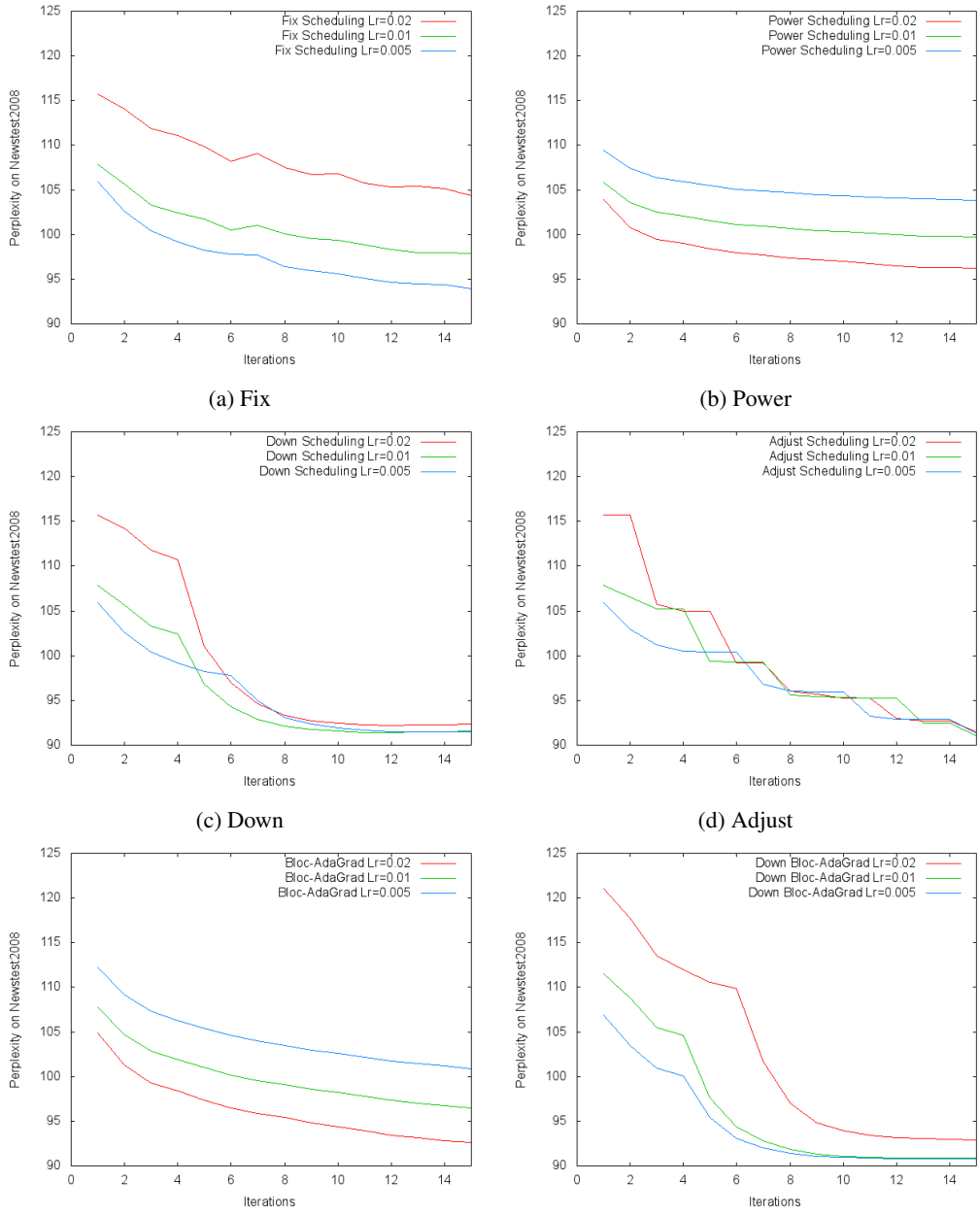
Tous les textes monolingues disponibles ont été utilisés pour l'apprentissage, soit 1.5 milliard de mots au total. L'apprentissage du réseau de neurone est itératif : à chaque itération, une sous-partie des données est échantillonnée aléatoirement, soit 15 millions de n-grammes dans nos expériences, puis divisée en paquet (mini-batch) de 128 n-grammes. Chaque paquet donne lieu à une mise à jour des paramètres du réseau. La perplexité est calculée à la fin de chaque itération sur le jeu de test *Newstest2008*. Afin de garantir que les différentes stratégies évaluées utilisent les mêmes données d'apprentissage, l'échantillon des données utilisé à chaque itération est conservé et partagé.

La procédure d'apprentissage du modèle SOUL est décrite dans (Le *et al.*, 2013) et résumée à la section 2.2. Les expériences suivantes concernent uniquement la dernière phase d'apprentissage des modèles SOUL, dont la configuration est la suivante : l'espace de projection est de dimension 500 ; le réseau contient deux couches cachées de dimensions respectives 1000 et 500 et la *short-list* contient les 2000 mots les plus fréquents.

4.1 Dépendance au choix des hyper-paramètres

Pour chaque stratégie d'ajustement, différents apprentissages ont été effectués en faisant varier la valeur initiale du pas d'apprentissage $\eta^{(0)}$. Les résultats en terme de perplexité sont représentés sur les figures 3a-3e. Sur ces courbes nous

6. http://www.statmt.org/wmt13/translation-task.html

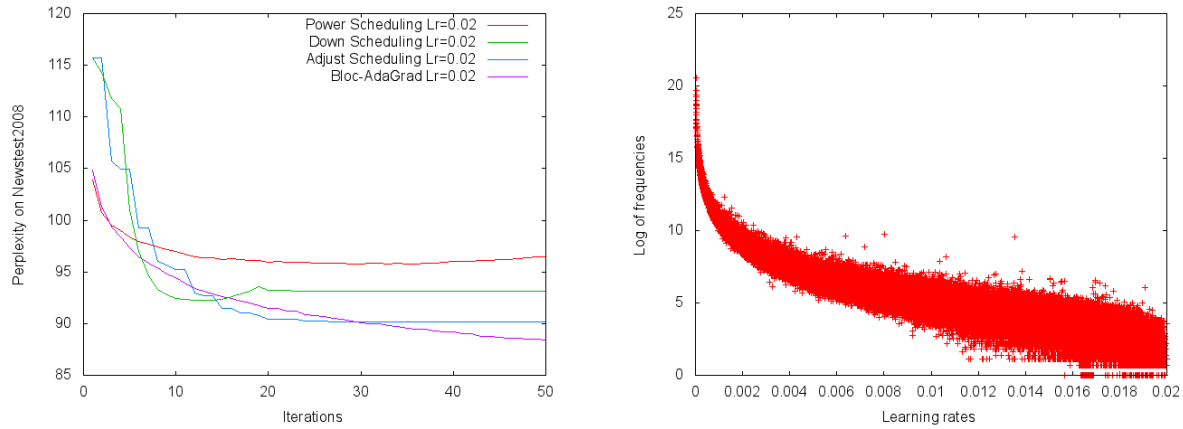


(a) Fix (b) Power

(c) Down (d) Adjust

(e) Bloc-AdaGrad (f) Down-Bloc-AdaGrad

FIGURE 3: Perplexités mesurées sur *Newstest2008* pour chacune des 6 stratégies décrites dans cet article. Dans le cas des figures 3c et 3f, chaque courbe d'apprentissage montre un point d'inflexion qui correspond à l'itération à laquelle le pas d'apprentissage est divisé par deux.



(a) Perplexités mesurées sur *Newstest2008* sur 50 itérations pour 4 stratégies d'ajustement. Pour la stratégie *Down*, on observe un phénomène de sur-apprentissage aux alentours de la 12ème itération.

(b) Relation entre la fréquence (échelle logarithmique) des mots et le pas d'apprentissage associé à leur représentation continue pour la stratégie *Bloc-AdaGrad*, après 40 itérations d'apprentissage.

FIGURE 4

pouvons observer que les performances obtenues avec les stratégies nommées *Fix*, *Power* and *Bloc-AdaGrad* sont très sensibles au choix de $\eta^{(0)}$, alors que les stratégies *Down* et *Adjust* mènent à des résultats plutôt stables au bout des 15 iterations, et ce avec différentes valeurs de $\eta^{(0)}$. Remarquons également que les deux dernières stratégies convergent nettement plus rapidement vers une perplexité basse.

De plus, dans le cas de la stratégie *Power*, la difficulté est augmentée par la présence d'un hyper-paramètre supplémentaire (le *learning rate decay*) τ. Dans toutes ces expériences, τ est fixé empiriquement à la valeur 5.10^{-7}. Néanmoins, cet hyper-paramètre a aussi une forte influence sur la vitesse de convergence.

Down-Bloc-AdaGrad : La forte dépendance au choix de η^0 de la méthode *AdaGrad* est un des défaut de cette méthode, puisque la recherche du paramètre optimal rend cette méthode coûteuse en terme de temps de calcul. Des travaux récents ont tenté de réduire cette dépendance, voir par exemple (Senior *et al.*, 2013). Appliqué au modèle SOUL, nous proposons une approche plus simple combinant la stratégie *Bloc-AdaGrad* avec la stratégie *Down*, qui est plus stable : d'une part, le gradient accumulé pour la première couche de sortie sert de valeur de normalisation, et d'autre part son influence est pondérée par le coefficient $\gamma^{(t)}$ dont la valeur est ajustée par la stratégie *Down*. Cette méthode est nommée *Down-Bloc-AdaGrad* : à l'instant t, la valeur du pas d'apprentissage pour le bloc b_j est calculée de la manière suivante :

$$\eta_{\mathrm{b}_j}^{(t)} = \frac{\gamma^{(t)} \times \left(1 + \mathrm{G}_{\mathrm{b}^*}^{(t)}\right)^{1/2}}{\left(1 + \mathrm{G}_{\mathrm{b}_j}^{(t)}\right)^{1/2}}$$

où b^* désigne le bloc de paramètres de la couche de sortie principale. Comme le montre la figure 3f, cette stratégie permet d'accélérer la vitesse de convergence tout en réduisant la dépendance au pas d'apprentissage initial, noté ici $\gamma^{(0)}$.

4.2 Apprentissage sur le long terme

Afin de comparer le comportement de différentes stratégies dans le cadre d'un apprentissage plus long, la figure 4a représente l'évolution de la perplexité au cours de 50 itérations d'apprentissage. Ces courbes suggèrent que seule la stratégie *Bloc-AdaGrad* peut tirer profit d'un apprentissage long, alors que les autres arrivent à stabilité à plus moins court terme. En particulier, les stratégies *Down* et *Power* donnent naturellement des courbes de perplexité qui se stabilisent rapidement, puisque le pas d'apprentissage tend rapidement vers 0. Au contraire, la stratégie *Bloc-AdaGrad* continue sa progression jusqu'au bout, puisque seuls les paramètres appartenant à un bloc fréquemment actif voient leurs pas d'apprentissage tendre vers 0. Ce phénomène concerne plus particulièrement les représentations continues des mots comme l'illustre la figure 4b. En effet les paramètres définissant la représentations continu d'un mot forment un bloc. Ainsi pour un mot fréquent, sa représentation continue sera fréquemment mise à jour et son pas d'apprentissage tendra rapidement vers 0,

comme l'exprime l'équation (8). Au contraire, les représentations continues des mots rares peuvent bénéficier d'un temps d'apprentissage plus long.

De plus, une autre considération complète l'analyse du comportement de la stratégie *Bloc-AdaGrad*. En autorisant différents pas d'apprentissage, cette stratégie donne une plus grande liberté à l'algorithme de descente de gradient pour choisir la direction à chaque étape. En effet, le pas d'apprentissage idéal à un instant donné doit tenir compte de la courbure de la fonction à optimiser : dans une zone à forte courbure, le gradient peut évoluer rapidement pour certains paramètres, il est donc alors préférable pour ces paramètres de réduire le pas d'apprentissage et d'effectuer des mises à jour fréquentes afin de pouvoir suivre ces variations rapides. Dans le cas d'une stratégie globale d'ajustement du pas d'apprentissage, les degrés de liberté manquent et, même si le pas d'apprentissage tend vers 0, il est impossible de nuancer le pas d'apprentissage par composante afin de suivre efficacement la fonction à optimiser. En dépit de ces propriétés, la différence en perplexité entre les stratégies globale *Bloc-AdaGrad* au bout de 50 itérations reste somme toute faible. Une interprétation est le peu de mots rares présent dans le corpus *Newstest2008*, alors que justement la particularité de la stratégie *Bloc-AdaGrad* réside dans la manière de traiter les mots rares. En pratique, il est rarement possible, ni souhaitable, d'utiliser un régime d'apprentissage long, bien sûr à cause du temps de calcul mais aussi car cela se révèle souvent inutile. Ainsi on observe qu'avec la stratégie *Bloc-AdaGrad*, en multipliant le temps d'apprentissage par 3, la perplexité baisse seulement de 3 points (comparaison entre l'itération 15 et 45 sur la figure 4a). Une telle différence de perplexité n'est susceptible de modifier qu'à la marge les performances d'un système de TAS ou de RAP.

La figure 4a justifie également pour la stratégie *Down* le recours à un apprentissage plus court (*early-stopping*) afin d'éviter le sur-apprentissage. On observe en effet que le modèle atteint sa plus basse perplexité aux alentours de la 10ème itération ; ensuite, on observe une évolution typique de sur-apprentissage : la perplexité mesurée lors de l'apprentissage (non représentée ici) continue de diminuer alors que celle mesurée sur un autre jeu de données commence à croître. Notons cependant qu'un apprentissage long, bien que coûteux, peut être souhaitable, afin de justement détecter, selon les stratégies employées, le phénomène de sur-apprentissage (Bengio, 2012). De plus, dans le cadre de certaines applications, les données d'apprentissage peuvent arriver au fil du temps, de manière « illimitée ». Dans ce cas, il est crucial de choisir une stratégie capable d'apprendre sur le long terme. À ce titre, la stratégie *Bloc-AdaGrad* semble la plus appropriée.

4.3 Impact sur les résultats en traduction automatique

Afin de mesurer l'impact de ces stratégies sur les performances d'un système de traduction automatique, certains des modèles sont utilisés au sein d'un système état de l'art pour la tâche de WMT13 de traduction de l'anglais vers l'espagnol. Étant donné le coût computationel des modèles de langue neuronaux, ces modèles sont utilisés en post traitement. Le système de traduction est utilisé dans un premier temps afin de générer les k meilleures hypothèses de traduction, $k = 300$ dans les expériences présentées dans cet article. Puis, dans un second temps le modèle de langue neuronal est utilisé afin d'évaluer chacune des hypothèses : le score produit par le modèle de langue neuronal est ajouté aux autres scores utilisés par le système de traduction ; puis les hypothèses sont retriées en utilisant des poids pour chacun des modèles qui ont été optimisés sur des données de développement (*Newstest2011* dans notre cas). Cette optimisation utilise l'algorithme KBMIRA (*k-best Batch Margin Infused Relaxed Algorithm*) décrit par Cherry & Foster (2012).

Les résultats sont regroupés dans le tableau 2. Les performances de traduction sont évaluées en terme de score BLEU mesuré sur l'ensemble de développement (*Newstest2011*) et deux jeux de données de test : *Newstest2012* et *Newstest2013*. Ces résultats montrent qu'il y a qu'une faible différence entre un modèle entraîné avec la stratégie *Bloc-AdaGrad* pendant 50 itérations et un modèle entraîné avec les deux stratégies *Down* et *Down-Bloc-AdaGrad* après 8 itérations. Ce résultat confirme l'intérêt d'utiliser un apprentissage abrégé si la stratégie d'ajustement est correctement choisie, et les stratégies du type *Down*, dont *Down-Bloc-AdaGrad* fait partie, semble les meilleures adaptées. De plus remarquons la dépendance marquée de la stratégie *Fix* au choix du pas d'apprentissage initial. Même si les différences ici sont plutôt mineurs, nous insistons sur le fait que notre intérêt dans le choix des stratégies se trouve dans le temps d'entraînement (le nombre d'itérations) et la stabilité du système par rapport au choix de paramètres (ici, le pas d'apprentissage initial).

5 Conclusions

Dans cet article, nous avons présenté une étude comparative de différentes stratégies d'ajustement du pas d'apprentissage utilisé par les modèles de langue neuronaux. Cet hyper-paramètre a, comme le montrent les résultats expérimentaux, une grande influence sur la vitesse de convergence et sur les performances du modèle. Comme cadre expérimental, nous

	Perplexité *Nt08*	dev *Nt11*	*Nt12*	*Nt13*
Sans modèle neuronal	-	32,32	33,86	29,79
Fix $\eta = 0,02$ ite. 15	104,4	32,80	34,12	29,97
Fix $\eta = 0,01$ ite. 15	97,8	32,93	34,30	30,18
Fix $\eta = 0,005$ ite. 15	93,9	32,98	34,37	30,20
Down $\eta^0 = 0,005$ ite. 8	93,1	33,01	34,35	30,11
Down-Bloc-AdaGrad $\gamma^{(0)} = 0,005$ ite. 8	91,4	33,05	34,43	30,06
Bloc-AdaGrad ite. 50 $\eta^0 = 0,02$	88,4	-	34,48	30,12

TABLE 2: BLEU scores mesurés sur *Newstest2012* et *Newstest2013*.

nous sommes intéressés à la tâche de traduction de WMT13, de l'anglais vers l'espagnol. Les modèles de langue neuronaux présentés utilisent l'architecture SOUL qui a pour caractéristique de pouvoir prendre en compte un vocabulaire de prédiction aussi grand que nécessaire. Outre les stratégies existantes, nous avons proposé deux nouvelles stratégies, nommées respectivement *Bloc-AdaGrad* et *Down-Bloc-AdaGrad*. Ces stratégies permettent de s'adapter au modèle étudié par leur capacité à ajuster le pas d'apprentissage, non pas de manière uniforme pour l'ensemble des paramètres, mais par groupe (ou bloc) de paramètres. Chacun de ces blocs regroupe les paramètres d'une même couche, ou ceux associés à la représentation continue d'un mot du vocabulaire.

Les résultats expérimentaux montrent que certaines des stratégies étudiées sont très sensibles au choix du pas d'apprentissage initial. De plus la méthode *Down-Bloc-AdaGrad* se distingue des autres en permettant d'obtenir une convergence rapide vers une perplexité basse, et ce, avec une insensibilité relative au choix du pas d'apprentissage initial. Dans de futurs travaux, nous pensons qu'utiliser cette stratégie dès la première étape d'apprentissage du modèle SOUL donnera des améliorations plus marquées.

Références

ALLAUZEN A., PÉCHEUX N., DO Q. K., DINARELLI M., LAVERGNE T., MAX A., LE H.-S. & YVON F. (2013). LIMSI @ WMT13. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, p. 62–69, Sofia, Bulgaria : Association for Computational Linguistics.

BENGIO Y. (2012). Practical recommendations for gradient-based training of deep architectures. In G. MONTAVON, G. B. ORR & K.-R. MÜLLER, Eds., *Neural Networks : Tricks of the Trade (2nd ed.)*, volume 7700 of *Lecture Notes in Computer Science*, p. 437–478. Springer.

BENGIO Y., DUCHARME R., VINCENT P. & JAUVIN C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, **3**(6), 1137–1155.

BOTTOU L. (2012). Stochastic Gradient Descent Tricks. In G. MONTAVON, G. ORR & K.-R. MÜLLER, Eds., *Neural Networks : Tricks of the Trade*, volume 7700 of *Lecture Notes in Computer Science*, p. 421–436. Springer Berlin Heidelberg.

BRANTS T., POPAT A. C., XU P., OCH F. J. & DEAN J. (2007). Large language models in machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, p. 858–867.

CHEN S. F. & GOODMAN J. T. (1998). *An Empirical Study of Smoothing Techniques for Language Modeling*. Rapport interne TR-10-98, Computer Science Group, Harvard University.

CHERRY C. & FOSTER G. (2012). Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 427–436 : Association for Computational Linguistics.

COLLOBERT R., WESTON J., BOTTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, **12**, 2493–2537.

DUCHI J., HAZAN E. & SINGER Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, **12**, 2121–2159.

GREEN S., WANG S., CER D. & MANNING C. D. (2013). Fast and adaptive online training of feature-rich translation models. In *Proceedings of the Annual Meeting on Association for Computational Linguistics (ACL)*, p. 311–321, Sofia, Bulgaria : Association for Computational Linguistics.

HUANG E., SOCHER R., MANNING C. & NG A. (2012). Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 873–882, Jeju Island, Korea : Association for Computational Linguistics.

KALCHBRENNER N. & BLUNSOM P. (2013). Recurrent continuous translation models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1700–1709, Seattle, Washington, USA : Association for Computational Linguistics.

LE H.-S., ALLAUZEN A. & YVON F. (2012a). Continuous space translation models with neural networks. In *Proceedings of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT)*, p. 39–48, Montréal, Canada : Association for Computational Linguistics.

LE H.-S., ALLAUZEN A. & YVON F. (2012b). Measuring the influence of long range dependencies with neural network language models. In *Proceedings of the NAACL-HLT 2012 Workshop : Will We Ever Really Replace the N-gram Model ? On the Future of Language Modeling for HLT*, p. 1–10, Montréal, Canada.

LE H.-S., OPARIN I., ALLAUZEN A., GAUVAIN J.-L. & YVON F. (2011). Structured output layer neural network language model. In *Proc. of ICASSP*, p. 5524–5527.

LE H. S., OPARIN I., ALLAUZEN A., GAUVAIN J.-L. & YVON F. (2013). Structured output layer neural network language models for speech recognition. *IEEE Transactions on Acoustic, Speech and Signal Processing*, **21**(1), 197 – 206.

LECUN Y., BOTTOU L., ORR G. & MULLER K. (1998). Efficient backprop. In G. ORR & M. K., Eds., *Neural Networks : Tricks of the trade* : Springer.

MIKOLOV T., KOMBRINK S., BURGET L., CERNOCKÝ J. & KHUDANPUR S. (2011). Extensions of recurrent neural network language model. In *Proc. of ICASSP*, p. 5528–5531.

MNIH A. & HINTON G. E. (2007). Three new graphical models for statistical language modelling. In *ICML*, p. 641–648.

MNIH A. & HINTON G. E. (2008). A scalable hierarchical distributed language model. In D. KOLLER, D. SCHUURMANS, Y. BENGIO & L. BOTTOU, Eds., *Advances in Neural Information Processing Systems 21*, volume 21, p. 1081–1088.

MNIH A. & TEH Y. W. (2012). A fast and simple algorithm for training neural probabilistic language models. In *Proceedings of the International Conference on Machine Learning*.

MORIN F. & BENGIO Y. (2005). Hierarchical probabilistic neural network language model. In *AISTATS'05*, p. 246–252.

NAKAMURA M., MARUYAMA K., KAWABATA T. & KIYOHIRO S. (1990). Neural network approach to word category prediction for english texts. In *Proceedings of the 13th conference on Computational linguistics (COLING)*, volume 3, p. 213–218.

OLLIVIER Y. (2013). Riemannian metrics for neural networks. *CoRR*, **abs/1303.0818**.

ROBBINS H. & MONRO S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, **22**, 400–407.

SCHAUL T., ZHANG S. & LECUN Y. (2012). No more pesky learning rates. preprint arXiv :1206.1106.

SCHWENK H. (2007). Continuous space language models. *Computer Speech and Language*, **21**(3), 492–518.

SENIOR A. W., HEIGOLD G., RANZATO M. & YANG K. (2013). An empirical study of learning rates in deep neural networks for speech recognition. In *Proc. of ICASSP*, p. 6724–6728.

SOCHER R., BAUER J., MANNING C. D. & ANDREW Y. N. (2013). Parsing with compositional vector grammars. In *Proceedings of the Annual Meeting on Association for Computational Linguistics (ACL)*, p. 455–465, Sofia, Bulgaria.

TURIAN J., RATINOV L.-A. & BENGIO Y. (2010). Word representations : A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 384–394, Uppsala, Sweden : Association for Computational Linguistics.

XU W. (2011). Towards optimal one pass large scale learning with averaged stochastic gradient descent. *CoRR*, **abs/1107.2490**.

YANG N., LIU S., LI M., ZHOU M. & YU N. (2013). Word alignment modeling with context dependent deep neural network. In *Proceedings of the Annual Meeting on Association for Computational Linguistics (ACL)*, p. 166–175, Sofia, Bulgaria.

Etude de l'impact de la translittération de noms propres sur la qualité de l'alignement de mots à partir de corpus parallèles français-arabe

Nasredine Semmar[1] Houda Saadane[2]

(1) Institut CEA LIST, DIASI, Laboratoire Vision et Ingénierie des Contenus, CEA Saclay – Nano-INNOV, 91191 Gif-sur-Yvette Cedex
(2) LIDILEM, Université Stendhal-Grenoble III, Domaine Universitaire, 1180, avenue centrale, 38400 Saint Martin d'Hères
nasredine.semmar@cea.fr, houda.saadane@e.u-grenoble3.fr

Résumé. Les lexiques bilingues jouent un rôle important en recherche d'information interlingue et en traduction automatique. La construction manuelle de ces lexiques est lente et coûteuse. Les techniques d'alignement de mots sont généralement utilisées pour automatiser le processus de construction de ces lexiques à partir de corpus de textes parallèles. L'alignement de formes simples et de syntagmes nominaux à partir de corpus parallèles est une tâche relativement bien maîtrisée pour les langues à écriture latine, mais demeure une opération complexe pour l'appariement de textes n'utilisant pas la même écriture. Dans la perspective d'utiliser la translittération de noms propres de l'arabe vers l'écriture latine en alignement de mots et d'étudier son impact sur la qualité d'un lexique bilingue français-arabe construit automatiquement, cet article présente, d'une part, un système de translittération de noms propres de l'arabe vers l'écriture latine, et d'autre part, un outil d'alignement de mots simples et composés à partir de corpus de textes parallèles français-arabe. Le lexique bilingue produit par l'outil d'alignement de mots intégrant la translittération a été évalué en utilisant deux approches : une évaluation de la qualité d'alignement à l'aide d'un alignement de référence construit manuellement et une évaluation de l'impact de ce lexique bilingue sur la qualité de traduction du système de traduction automatique statistique Moses. Les résultats obtenus montrent que la translittération améliore aussi bien la qualité de l'alignement de mots que celle de la traduction.

Abstract. Bilingual lexicons play a vital role in cross-language information retrieval and machine translation. The manual construction of these lexicons is often costly and time consuming. Word alignment techniques are generally used to construct bilingual lexicons from parallel texts. Aligning single words and nominal syntagms from parallel texts is relatively a well controlled task for languages using Latin script but it is complex when the source and target languages do not share the same written script. A solution to this issue consists in writing the proper names present in the parallel corpus in the same written script. This paper presents, on the one hand, a system for automatic transliteration of proper names from Arabic to Latin script, and on the other hand, a tool to align single and compound words from French-Arabic parallel text corpora. We have evaluated the word alignment tool integrating transliteration using two methods: A manual evaluation of the alignment quality and an evaluation of the impact of this alignment on the translation quality by using the statistical machine translation system Moses. The obtained results show that transliteration of proper names from Arabic to Latin improves the quality of both alignment and translation.

Mots-clés : Lexique bilingue, translittération, alignement de mots, traduction automatique statistique, évaluation.
Keywords: Bilingual lexicon, transliteration, word alignment, statistical machine translation, evaluation.

1 Introduction

Les lexiques bilingues jouent un rôle important dans les applications de Traitement Automatique des Langues (TAL) telles que la Recherche d'Information Interlingue (RII) et la Traduction Automatique (TA). La construction manuelle de ces lexiques est lente et coûteuse. C'est la raison pour laquelle depuis quelques années de nombreux travaux ont fait appel aux techniques d'alignement pour automatiser le processus de construction de lexiques bilingues. Ces travaux ont montré que l'alignement de formes simples et de syntagmes nominaux à partir de corpus parallèles est une tâche relativement bien maîtrisée pour les langues à écriture latine. En revanche, l'appariement de textes parallèles n'utilisant pas la même écriture demeure une opération complexe. Ce qui a conduit plusieurs chercheurs à exploiter la transcription

ou la translittération de certains mots des textes parallèles comme « points d'ancrage » pour améliorer la mise en correspondance bilingue. La transcription consiste à substituer à chaque son ou à chaque phonème d'un système phonologique, un graphème ou un groupe de graphèmes d'un système d'écriture, tandis que la translittération consiste à substituer à chaque graphème d'un système d'écriture un autre graphème ou un groupe de graphèmes d'un autre système d'écriture, indépendamment de la prononciation.

Dans la perspective d'évaluer l'impact de l'utilisation de la translittération de noms propres sur la qualité d'un lexique bilingue français-arabe construit automatiquement, nous présentons dans cet article, d'une part, un système de translittération de noms propres de l'arabe vers l'écriture latine et un outil d'alignement de mots simples et composés à partir de corpus de textes parallèles français-arabe, et d'autre part, les résultats d'évaluation de ce lexique bilingue selon deux approches (intrinsèque et extrinsèque) et utilisant deux corpus différents (ARCADE II et OPUS).

La suite de l'article est organisée comme suit : dans la section 2, nous présentons l'approche de la translittération des noms propres écrits en arabe vers l'écriture latine. Puis nous décrivons dans la section 3, l'outil d'alignement de mots à partir d'un corpus de textes parallèles français-arabe en nous focalisant plus particulièrement sur l'étape d'appariement de cognats qui exploite la translittération. La section 4 sera consacrée aux expérimentations effectuées ainsi que la présentation des résultats obtenus et la section 5 conclut notre étude et présente nos travaux futurs.

2 Translittération

Les évolutions rapides des nouvelles technologies d'information et de communication sont accompagnées d'un essor important de la quantité et la diversité d'information générée et manipulée notamment celle disponible sur le web. Cette dernière, étant destinée à un public large et varié, est transcrite dans différentes langues ce qui a fait émerger la nécessité d'internationaliser les contenus afin de permettre un partage de données le plus large possible, entre des utilisateurs manipulant des langues différentes. Ainsi, les techniques de translittération trouvent tout leur intérêt afin de rendre cette perspective de partage possible.

2.1 Etat de l'art

Plusieurs travaux de recherche sur la transcription et la translittération ont été menés ces dernières années. Nous citons à titre d'exemple les travaux de (Jiang et al., 2007) pour la translittération des entités nommées (ENs) du chinois vers l'anglais, qui utilisent un modèle d'entropie maximale pour déterminer la translittération candidate, en se basant sur la similarité phonétique avec l'EN dans la langue source. Ces méthodes fonctionnent bien avec les entités nommées qui sont traduites phonétiquement, mais ce n'est pas toujours le cas. Pour ce type d'ENs, il est plus recommandé d'explorer les similitudes sémantiques entre les ENs dans les différentes langues. Ce constat a été approuvé dans les travaux de (Huang et al., 2004) qui combine les similitudes sémantiques et phonétiques. Les expérimentations effectuées montrent que cette approche réalise une précision de 67%. Par ailleurs, (Huang et al., 2003) ont travaillé sur l'extraction des paires d'ENs hindi-anglais grâce à l'alignement d'un corpus parallèle. Des paires chinois-anglais sont d'abord extraites à l'aide d'une programmation dynamique. Ce modèle chinois-anglais est alors adapté à l'hindi-anglais de manière itérative, en utilisant les paires hindi-anglais d'entités nommées déjà extraites pour l'amorçage du modèle. On trouve aussi des propositions de systèmes visant à attribuer une seule translittération à un nom donné : c'est le cas du modèle génératif proposé pour les noms d'origine anglaise écrits en japonais vers le système d'écriture latin (Knight, Graehl, 1997). Cette approche a été adaptée par (Stalls, Knight, 1998) à la façon dont un nom anglais écrit en arabe est transcrit en anglais. Le système de génération de translittérations s'appuie sur un dictionnaire d'apprentissage et ne prend pas en compte les prononciations non répertoriées ou inconnues du dictionnaire. Pour pallier cette limitation, certains travaux utilisent un modèle non supervisé. C'est le cas du système de translittération des noms anglais vers l'arabe proposé par (Abduljaleel, Larkey, 2003). Ce système est fondé sur le calcul de la forme la plus probable, censée être la forme correcte. Or cette hypothèse n'est pas vérifiée pour tous les pays arabes ni pour tous les dialectes. (Alghamdi, 2005) a proposé un système de translittération en écriture anglaise des noms arabes voyellés pour contourner la difficulté de la prononciation et le problème des variantes dialectales. Ce système est basé sur un dictionnaire de noms arabes dans lequel la prononciation est réglée au moyen de voyelles ajoutées aux noms répertoriés, avec indication en vis à vis de leur équivalent en écriture anglaise. Cependant, cette approche, non seulement ne prend pas en compte les prononciations non répertoriées dans le dictionnaire, mais, de plus, elle est normative par le fait qu'elle ne propose qu'une seule translittération pour un nom donné.

En conclusion, la plupart des travaux actuels ne prennent pas en compte la complexité du problème de la transcription et de la translittération qui concerne aussi bien l'oralité que le modèle scriptural des systèmes linguistiques impliqués. En effet, très peu de travaux prennent en considération le lien entre phonologie comparée et transcription interlingue, entre

graphématique comparée et translittération multilingue et entre dialectologie arabe et systèmes de translittération latins. Les rares études qui proposent une solution prenant en compte partiellement l'une de ces problématiques sont dédiées à l'identification automatique de l'origine du locuteur à partir de son dialecte (Guidère, 2004) (Barkat-Defradas et al., 2004). Dans le cadre de cette étude, notre objectif est de proposer un système automatique de translittération qui tient compte du lien entre phonologie, graphématique et dialectologie, dans la transcription des noms et des prénoms arabes vers l'écriture latine et plus particulièrement pour le français et l'anglais (Pouliquen, Steinberger, 2007).

2.2 Approche proposée pour la translittération de noms propres de l'arabe vers l'écriture latine

Afin de renvoyer la totalité des cas possibles de la translittération d'un nom arabe en écriture latine, nous nous sommes intéressés aux questions et aux problèmes liés à la translittération basée sur le système phonétique de l'arabe littéraire ainsi que sur la majorité des familles de dialectes, en prenant en compte des nombreuses variantes régionales et locales. Nous avons commencé par recenser les translittérations existantes pour chaque lettre de l'alphabet arabe standard depuis les normes et usages observés sur le Web et sur les dictionnaires de lieux géographiques de GeoNames. Nous avons constaté qu'au sein du même dictionnaire géographique un nom propre peut avoir plusieurs translittérations différentes. Cette investigation empirique est basée sur un corpus de textes qui a été recueilli dans les différentes langues cibles visées par le translittérateur. Elle a permis de constituer une librairie des équivalents graphématiques utilisés dans les écrits utilisant l'alphabet latin. Ci-dessous quelques équivalences graphématiques établies à partir de cette étude sur différents corpus :

- La lettre ش est transcrite en S dans DIN-31635, Sh selon UN, EI & ALA-LC, š suivant ISO/R 233 et (ch) dans le corpus d'apprentissage.
- La lettre ظ est transcrite en ẓ dans les différentes normes de translittération et en z, dh et d dans le corpus d'apprentissage.

Nous avons défini un certain nombre de règles syntaxiques et contextuelles afin de recenser les différentes translittérations. Parmi les règles syntaxiques que nous avons considérées dans notre translittération, le fait que le nom arabe ne prend pas en compte la dernière voyelle courte ou tanwin (marqueur du cas) à la fin du mot. Par exemple : زار بلال محمداً, le prénom محمداً est transcrit par Mohammed et non pas Mohammedan. Le module de translittération de l'écriture arabe vers l'écriture latine tient compte du lien entre la phonologie, la graphématique et la dialectologie en utilisant un certain nombre de règles issues d'une étude expérimentale. Il est fondé sur les automates à états finis pondérés de type transducteurs. Nous avons utilisé l'outil HTFST qui est constitué d'une interface basée sur la librairie open-source OpenFst (Reley et al., 2009). Cet outil sert à créer les automates de règles morphologiques, syntaxiques, et autres, et les appliquer ensuite à des textes. HTFST possède aussi une syntaxe propre aux « règles de replacements parallèles et contextuelles » offrant les mêmes possibilités que celles de XFST (Xerox Finite State Tool) (Beesley, Karttunen, 2003) implémentées en utilisant la librairie FOMA (Hulden, 2009). Le fonctionnement de notre approche de translittération est déterminé par la nature du mot fourni en entrée : l'automate passe d'état en état suivant les transitions, à la lecture de chaque lettre arabe de l'entrée. La Figure 1 décrit l'organigramme de notre module de translittération:

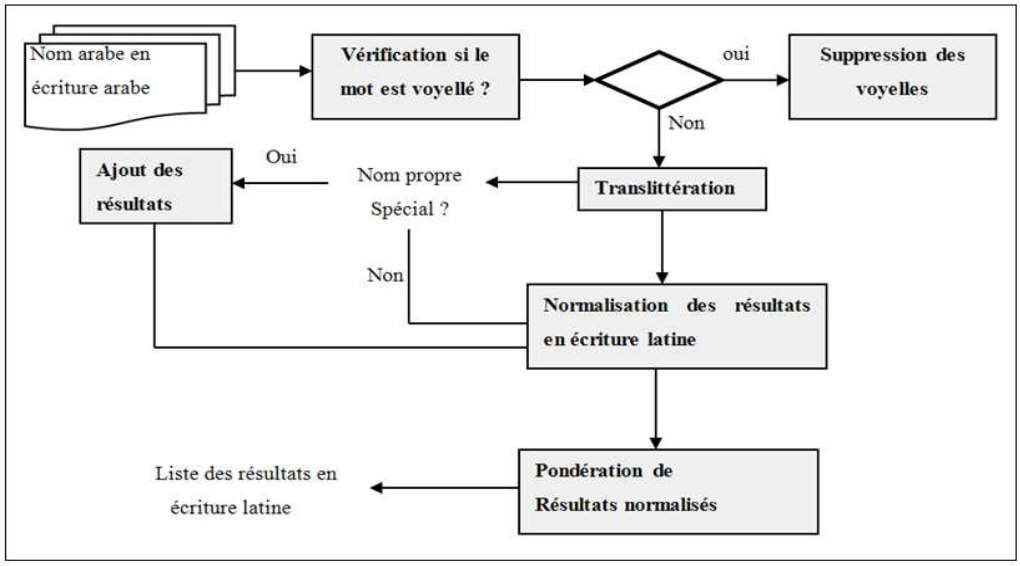


FIGURE 1: Organigramme du fonctionnement du translittérateur de l'arabe vers le latin

A l'issue de la lecture, un premier automate traite l'entrée de la manière suivante: si l'entrée est voyellée, il supprime les voyelles avant de translittérer le nom; si l'entrée est non-voyellée, il procède directement à la translittération du nom. Nous supprimons les voyelles afin de générer toutes les translittérations françaises et anglaises possibles. Ceci est dû à l'influence des dialectes sur les voyelles où les translittérations des mots issus du dialecte du Macherek sont orientées vers la translittération anglaise et ceux du dialecte du Maghreb sont plus orientées vers la translittération française. Enfin, le module produit en sortie une liste triée de noms arabes écrits en caractères latins.

Le cœur du système de translittération est constitué de règles contextuelles qui permettent le remplacement des lettres arabes en lettres latines ainsi que l'ajout des voyelles latines, en prenant en compte les lettres situées devant et/ou derrière la lettre à ajouter ou remplacer. Ces règles visent aussi à rendre compte de la manière la plus précise possible des formes observées en entrée : S'agit-il d'une «kunya» ? D'un nom précédé d'un article ? Ou bien d'un prénom seul ? Selon la forme d'entrée, nous appliquons d'abord des règles adéquates pour transcrire la partie qui ne constitue pas le nom à proprement parler (particules). Ainsi, des noms propres (spéciaux) comme (Ibn) إبن, (Abd) عبد, (Taha) طه, etc. seront transcrits directement. Prenons, par exemple, le prénom عبد qui peut être translittéré de plusieurs façons différentes. Nous attribuons un poids pour chaque translittération, sachant que le poids le plus bas indique la solution la plus probable. La règle ci-dessous indique que, lorsqu'un mot débute par ع ب د, il est transcrit le plus souvent par « Abd », ou bien moins souvent par « Abed ». Plus rarement, il sera transcrit « 3abd » ou « 3Abd », et dans quelques cas il sera transcrit par « Abd ».

R = ((ع ب د) .x. (((\` A b d <3000>) | A b d | (A b e d <1000>) | (3 (a|A) b d <2000>))).*;

Après avoir traité la partie (spéciale) du nom propre, nous appliquons les règles pour la translittération des noms eux-mêmes. Les règles pour la translittération des noms s'appliquent à leur tour selon le nombre de consonnes du nom considéré, et dans un ordre de priorité déterminé.

Par exemple, pour translittérer en écriture latine le nom propre arabe عبد الرشيد qui est composé par Abd (عبد) + Al (ال) + Nom (رشيد), le système procède de la manière suivante :

- Translittération de la particule عبد «Abd»;
- Translittération de l'article ال «Al»;
- Concaténation de la particule «Abd» et de l'article «Al» en les reliant au nom par un trait d'union ou en insérant un blanc entre les deux : Abd Al Rachid (عبد الرشيد);
- Génération de toutes les formes de translittération possibles pour ces trois éléments (Table 1):

Nom propre arabe	Translittérations
عبد الرشيد	Abd Al-Rachid
	Abdul Rashid
	abd al-Rashid
	3abd El Rachid
	abd Al Rashid
	Abdar-Rashid
	Abdel Rachid

TABLE 1 : Quelques translittérations pour le nom propre composé «عبد الرشيد»

- Normalisation de la liste des noms en écriture latine en supprimant les caractères spéciaux (diacritiques et chiffres) et en ajoutant la majuscule au début de chaque nom propre;
- Pondération de la liste des noms en écriture latine en attribuant un poids aux règles qui ont servi à la génération de la liste. Cette pondération est réalisée en utilisant divers moteurs de recherche en notant à chaque fois le nombre d'occurrences pour chaque forme générée du nom propre.

3 Alignement de mots

L'alignement de mots ou l'extraction de lexiques bilingues à partir de corpus de textes parallèles peut se décomposer conceptuellement en deux aspects: il s'agit de repérer les mots du texte source et du texte cible, puis de les mettre en correspondance.

3.1 Etat de l'art

Il existe principalement trois approches pour l'alignement de mots à partir de corpus de textes parallèles alignés phrase à phrase:

- Les approches à dominante statistique qui s'appuient sur les modèles IBM (Brown et al., 1993). L'outil d'alignement GIZA++ (Och, Ney, 2000) implémente notamment ce type d'approche. Cet outil implémente divers modèles de traduction (IBM 1, 2, 3, 4, 5 et HMM). GIZA++ est un outil efficace pour aligner les mots simples, mais il est moins performant, d'une part, lorsque les langues source et cible ont des morphologies et des structures syntaxiques différentes, et d'autre part, pour aligner les expressions multimots (Allauzen, Wisniewski, 2009) (Abdulhay, 2012).
- Les approches linguistiques qui utilisent généralement des dictionnaires bilingues déjà disponibles mais aussi les résultats de l'analyse morpho-syntaxique des phrases source et cible (Debili, Zribi, 1996). Les méthodes proposées par (Debili, Zribi, 1996) utilisent des ressources linguistiques externes (lexiques, règles, etc.) pour apparier les mots des textes parallèles alignés au niveau de la phrase. Ces méthodes font l'hypothèse que pour que des phrases soient en correspondance de traduction, il faut que les mots qui les composent soient également en correspondance. Elles n'utilisent qu'une information interne, c'est-à-dire que toute l'information nécessaire (et en particulier les correspondances lexicales) est dérivée des textes à aligner eux-mêmes (ancrage lexical).
- Une combinaison des méthodes statistiques avec différentes sources d'information linguistique (Daille et al., 1994) (Gaussier, Langé, 1995) (Ozdowska, Claveau, 2006) (Semmar et al., 2010). La méthode proposée par Gaussier (1995) est fondée sur des modèles statistiques pour établir les associations entre mots anglais et mots français, et ce en exploitant la propriété de dépendance entre les mots et leurs traductions respectives. La prise en compte des positions des mots dans les phrases permet de constituer un modèle de distorsion qui aide à la construction des associations. Ensuite, les structures morpho-syntaxiques représentant les séquences admissibles d'étiquettes grammaticales et de mots ont été recensées. Les correspondances et non-correspondances entre les structures anglaises et françaises sont utilisées pour élaborer les modèles statistiques permettant de retrouver les équivalences entre termes anglais et termes français. Quant à l'approche développée par Ozdowska et Claveau (2006), elle consiste d'abord à apparier les mots à un niveau global grâce au calcul des fréquences de cooccurrence dans des phrases alignées. Ensuite, ces mots constituent les couples amorces qui servent de point de départ à la propagation des liens d'appariement à l'aide des différentes relations de dépendance identifiées par un analyseur syntaxique dans chacune des deux langues.

Contrairement à l'alignement de mots simples qui est désormais une tâche bien maîtrisée plus particulièrement pour les langues à écriture latine, l'alignement d'expressions multimots continue à susciter de nombreux travaux de recherche (Ozdowska, Claveau, 2006) (Lefever et al., 2009) (Bouamor et al., 2012). La plupart de ces travaux commencent tout d'abord par identifier les expressions multimots dans chaque partie du corpus parallèle, ensuite, utilisent différentes approches d'alignement pour les apparier. Les approches pour l'extraction monolingue d'expressions multimots peuvent être: (1) symboliques en reposant sur des patrons morpho-syntaxtiques (Okita et al., 2010), (2) statistiques en utilisant des mesures d'association pour classer les expressions multimots candidates (Vintar, Fisier, 2008), et (3) hybrides combinant (1) et (2) (Seretan, Wehrli, 2007). Pour identifier les correspondances entre expressions multimots dans différentes langues, plusieurs travaux font appel à des outils d'alignement de mots simples pour guider l'alignement d'expressions multimots. D'autres se basent sur des algorithmes d'apprentissage statistique. Une hypothèse largement suivie pour acquérir des expressions multimots bilingues est qu'une expression multimots dans une langue source garde la même structure syntaxique que son équivalente dans une langue cible donnée (Seretan, Wehrli, 2007) (Tufis, Ion, 2007). Or, cette hypothèse n'est pas toujours vérifiée puisque certaines expressions multimots ne se traduisent pas forcément par des expressions ayant la même structure syntaxique. De même, certaines expressions ne se traduisent pas systématiquement par une expression de même longueur.

Pour les langues n'utilisant pas l'écriture latine, de nombreux travaux ont été réalisés pour aligner automatiquement les translittérations à partir de corpus de textes multilingues en vue de l'enrichissement de lexiques bilingues. Citons notamment les travaux de (Yaser, Knight, 2002) et (Sherif, Kondrak, 2007) sur l'alignement arabe-anglais, (Tao et al., 2006) sur l'utilisation de la translittération pour l'extraction d'entités nommées à partir de corpus comparables ainsi que (Shao, Ng, 2004) qui utilisent l'information apportée par les translittérations sur la base de leur prononciation. Ils combinent l'information apportée par le contexte des traductions avec l'information apportée par les translittérations entre l'anglais et le chinois. L'intérêt de ce travail réside dans le fait qu'il permet l'alignement de mots très spécifiques mais rares.

Nous décrivons, dans la section suivante, notre démarche pour extraire un lexique bilingue de mots simples et de mots composés à partir d'un corpus parallèle français-arabe aligné au niveau de la phrase.

3.2 Approche proposée pour l'alignement de mots à partir de corpus de textes parallèles français-arabe

La démarche que nous proposons pour la construction de lexiques bilingues à partir de corpus de textes parallèles, est composée des trois étapes suivantes:

- alignement de mots simples,
- alignement de mots composés se traduisant mot à mot,
- alignement d'expressions multimots.

Notre approche pour l'alignement de mots est basée, d'une part, sur un modèle linguistique utilisant un dictionnaire bilingue, les caractéristiques des cognats, les catégories grammaticales, les relations de dépendance syntaxique et les règles de reformulation pour l'alignement de mots simples et composés, et d'autre part, sur un modèle hybride combinant patrons morpho-syntaxiques et méthodes statistiques pour l'alignement d'expressions multimots. Les entrées de l'outil d'alignement, implémentant notre approche, sont les sorties normalisées d'une analyse morpho-syntaxique effectuée à l'aide de la plate-forme d'analyse linguistique LIMA (Besançon et al., 2010) sur le corpus de textes parallèles. Cette plate-forme fournit pour chaque couple de phrases source et cible :

- la liste des lemmes et des formes fléchies des mots ainsi que leur position dans la phrase,
- les catégories grammaticales des mots,
- les relations de dépendance syntaxique entre les mots et les mots composés.

Le processus de normalisation consiste à supprimer les mots vides de la liste des lemmes des mots retournés par la plate-forme LIMA. Les mots vides sont identifiés à partir de leur catégorie grammaticale (prépositions, articles, ponctuations et certains adverbes). Nous considérons les mots restants comme des mots significatifs (pleins).

Nous décrivons ci-dessous uniquement les principaux modules composant l'aligneur de mots simples et nous nous focalisations sur l'étape qui concerne l'alignement de mots utilisant la détection de cognats et d'entités nommées dans les phrases source et cible. C'est cette étape qui utilise la translittération des noms propres de l'arabe vers l'écriture latine. Les modules d'alignement de mots composés et d'expressions multimots sont décrits respectivement dans (Semmar et al., 2010) et (Bouamor et al., 2012). L'alignement de mots simples se déroule selon les trois étapes suivantes:

- alignement utilisant le dictionnaire bilingue préexistant,
- alignement utilisant la détection de cognats et d'entités nommées dans les phrases source et cible,
- alignement utilisant les catégories grammaticales des mots des phrases source et cible.

L'alignement en utilisant le dictionnaire bilingue préexistant consiste, d'une part, à extraire les traductions des lemmes significatifs des phrases de la langue source en interrogeant le dictionnaire bilingue, et d'autre part, à rechercher la traduction dans la phrase cible et en comparant sa position avec celle du lemme à aligner. Si les positions des deux lemmes source et cible sont dans une même fenêtre de taille *n* respectivement dans les phrases source et cible, alors ils seront considérés traduction l'un de l'autre. Nous avons fixé expérimentalement la valeur de *n* à 6. Ainsi, le mot de la phrase source Mot_{source} est considéré comme traduction du mot de la phrase cible Mot_{cible} si les conditions [1] et [2] sont vérifiées :

$$Position\ (Mot_{source}) - 3 <= Position\ (Mot_{cible}) \quad [1]$$

$$Position\ (Mot_{cible}) <= Position\ (Mot_{source}) + 3 \quad [2]$$

Nous avons constaté aussi que beaucoup de noms arabes ne sont pas reconnus comme entités nommées par la plate-forme LIMA. Cela vient du fait que cette plateforme utilise des listes ainsi que des règles de déclencheurs pour reconnaître des entités telles que les noms de personnes, d'organisations, de lieux… mais ces listes sont limitées et plus particulièrement pour les langues peu dotées comme l'arabe. C'est pour cette raison que nous avons ajouté une étape supplémentaire à notre outil d'alignement de mots simples. Cette étape est utilisée pour permettre l'appariement des cognats présents dans les phrases source et cible. En linguistique, les cognats sont des paires de mots de langues différentes qui partagent des propriétés phonologiques, orthographiques et sémantiques. Nous pouvons étendre cette définition aux noms propres et aux expressions numériques puisqu'ils varient en général légèrement d'une langue à une autre. Plusieurs travaux ont montré que la détection et la mise en correspondance des cognats dans les textes source et cible permettent d'améliorer les résultats d'alignement au niveau des phrases (Simard et al., 1993) mais aussi des mots (Al-Onaizan, Knight, 2002) (Kondrak, 2005) (Kraif, 2001). Récemment, Frunza et Inkpen (2009) ont évalué une

méthode qui utilise 13 mesures de similarité orthographique pour identifier les cognats et les « faux amis ». Nous considérons dans une première étape comme cognats les mots dont les quatre premiers caractères sont identiques. Cette étape est simple à implémenter lorsque les phrases source et cible sont écrites avec le même script ou dans deux scripts proches. Dans notre étude, l'alignement de mots est réalisé à partir de corpus de textes parallèles français-arabe. Or ces deux langues sont écrites avec deux scripts différents. Pour détecter les cognats présents dans ces textes, nous avons utilisé le système de translittération décrit précédemment pour transformer les noms propres écrits en arabe vers l'écriture latine. Cette première étape a permis de détecter que les noms propres « Garner » et « Irak » et leur translittération respective en écriture latine «garnir» (du nom propre « غارنر ») et « irak » (du nom propre «العراق») sont des cognats. En revanche, cette étape ne permet pas d'aligner des couples de mots comme « Algérie » et « aljezeyr » (translittération du nom propre « الجزائر »). Pour ce faire, nous avons utilisé la distance Jaro–Winkler (Winkler, 1990), une mesure de similarité basée sur le nombre de lettres en commun entre le mot de la langue source *ms* et le mot de la langue cible *mc*.

$$DJ(ms,mc) = \begin{cases} 0 & si\ m = 0 \\ \frac{1}{3}\left(\frac{m}{|ms|} + \frac{m}{|mc|} + \frac{m-t}{m}\right) & sinon \end{cases}$$

Où:

- *m* est le nombre de caractères correspondants. Deux caractères identiques des mots *ms* et *mc* sont considérés comme correspondants si leur éloignement (la différence entre leurs positions dans leurs chaînes respectives) ne dépasse pas :

$$\left(\frac{max(|ms|,|mc|)}{2}\right) - 1$$

- *t* est le nombre de transpositions. Ce nombre est obtenu en comparant le i^{ème} caractère correspondant du mot *ms* avec le i^{ème} caractère correspondant du mot *mc*. Le nombre de fois où ces caractères sont différents, divisé par deux, donne le nombre de transpositions.
- |*ms*|, |*mc*| correspondent aux longueurs en nombre de caractères des mots *ms* et *mc*.

La mesure de similarité Jaro–Winkler est une variante de la distance Jaro *DJ* (Jaro, 1989).

$$DJW(ms,mc) = DJ(ms,mc) + (lp(1 - DJ(ms,mc)))$$

Où *l* est la longueur du préfixe commun et *p* est un coefficient qui permet de favoriser les chaînes avec un préfixe commun.

Pour fixer les valeurs de *l* et *p* ainsi que le seuil pour lequel deux mots sont considérés comme cognats, nous avons utilisé un échantillon de 100 noms propres arabes translittérés en écriture latine. Dans cet échantillon, un nombre propre écrit en arabe peut avoir en moyenne 37 translittérations en écriture latine mais il existe des noms propres qui peuvent dépasser les 1 000 translittérations comme c'est le cas du mot « الجزائر » (Algérie) qui en a 1 120. Nous avons constaté que les valeurs de *l* et *p* qui permettent d'accepter le plus grand nombre de translittérations pour un nom propre sont respectivement 2 et 0,1 pour un seuil de cognats égal à 0,9. Ces paramètres fixés empiriquement permettent certes d'identifier comme cognats le mot «Algérie» et la translittération «aljezeyr» mais génèrent aussi des erreurs puisque cet aligneur considère par exemple que les mots «mohamed» et la translittération «mahmoud» du nom propre arabe «محمود» sont des cognats. Pour réduire ce type d'erreurs, nous vérifions les conditions [1] et [2] relatives aux positions des mots respectivement dans les phrases source et cible.

Certes, la détection de cognats améliore significativement les résultats de l'alignement mais ça concerne uniquement les corpus de textes ayant une forte présence de noms propres. Pour détecter de nouvelles correspondances, nous prenons en compte les paires de mots des langues source et cible qui ont les mêmes catégories grammaticales et dont les positions vérifient les conditions [1] et [2] décrites précédemment. Cette étape est particulièrement performante pour identifier les traductions des mots entourés par des mots déjà traduits.

Le tableau ci-dessous (Table 2) présente le résultat de l'alignement de mots simples et de mots composés se traduisant mot à mot de la phrase source « Le général Garner a laissé entendre que l'occupation de l'Irak ne serait pas éternelle. » et de sa traduction en langue cible «اشار الجنرال غارنر الى ان احتلال العراق لن يدوم الى الابد.».

Lemmes des mots de la phrase en langue source	Lemmes des mots de la phrase en langue cible	Etape d'alignement utilisée
général	جنرَال	Appariement de catégories grammaticales
Garner	غارنر	Appariement de cognats
laisser	أشَارَ	Appariement de catégories grammaticales
occupation	إحْتِلاَل	Dictionnaire bilingue
Irak	العِرَاق	Appariement de cognats
général_garner	جنرَال_غارنر	Mise en correspondance de mots composés
occupation_Irak	إحْتِلاَل_العِرَاق	Mise en correspondance de mots composés

TABLE 2 : Résultat de l'alignement de mots simples et composés

Ce tableau montre, d'une part, que les lemmes « entendre », « être » et « éternel » de la phrase source n'ont pas été alignés, et d'autre part, que l'alignement du lemme « laisser » n'est pas correct. En vérifiant dans le dictionnaire bilingue, nous avons trouvé plusieurs traductions pour ces lemmes, mais ils n'ont pas été alignés car ces traductions ne sont pas présentes dans la phrase cible. Cet exemple montre bien l'intérêt des alignements n:m (dans notre exemple il s'agit d'un alignement 2:1 pour le lemme « laisser entendre » qui aurait du être aligné avec le lemme «أشَارَ») même s'ils ne sont pas aussi fréquents que les alignements 1:1. Notons que le lexique bilingue construit à l'issue du processus d'alignement de mots contient les alignements corrects et incorrects, mais, les lemmes qui n'ont pas été alignés ne seront pas pris en compte. Les symboles « _ » séparant les lemmes des mots composés seront remplacés par des espaces.

4 Résultats expérimentaux et discussion

Pour illustrer l'apport de la translittération sur la qualité du lexique bilingue produit par l'alignement de mots simples et composés, nous avons évalué les résultats de l'alignement selon deux approches différentes :

- une évaluation manuelle comparant les résultats de notre aligneur de mots par rapport à un alignement de référence,
- une évaluation automatique en intégrant les résultats de notre aligneur de mots dans le corpus d'apprentissage du modèle de traduction du système de traduction statistique libre Moses (Koehn et al., 2007).

L'évaluation manuelle de l'aligneur de mots a été réalisée sur une partie composée de 1 000 phrases du corpus MD (Monde Diplomatique) français-arabe de la campagne ARCADE II (Véronis et al., 2008). Cet alignement de référence au niveau des mots simples et composés a été construit manuellement à l'aide de l'outil Yawat (Germann, 2008). Pour les métriques d'évaluation, nous avons utilisé celles du protocole défini lors de la conférence HLT/NAACL 2003 (Mihalcea, Pedersen, 2003). La table 3 résume nos résultats en termes de précision et de rappel selon que l'aligneur de mots utilise ou non l'appariement de cognats avec la translittération de noms propres. Ces résultats montrent que l'utilisation de la translittération arabe permet d'augmenter aussi bien la précision que le rappel et confirment les résultats que nous avons obtenus précédemment sur un petit corpus de 283 phrases (Saadane, Semmar, 2012) ainsi que ceux de (Kondrak et al., 2003) qui ont pu réduire de 10% le taux d'erreurs de l'alignement de mots en utilisant l'appariement de cognats. Le lexique bilingue extrait à partir des 1 000 paires de phrases en utilisant notre outil d'alignement de mots contient 16 291 entrées dont 2 023 noms propres. L'analyse de ce lexique montre qu'il contient un nombre important de doublons plus particulièrement pour les noms propres mais aussi quelques traductions de mots polysémiques. En outre, environ 53% des mots alignés se trouvaient dans le dictionnaire bilingue et 12% ont été alignés à l'aide du module d'appariement de cognats qui utilise la translittération.

Alignement de mots	Précision	Rappel	F-Mesure
sans l'appariement de cognats (sans translittération)	0,82	0,86	0,83
avec l'appariement de cognats (avec translittération)	0,87	0,88	0,87

TABLE 3 : Résultats de l'évaluation de l'alignement de mots

L'évaluation automatique de notre aligneur de mots a été réalisée en utilisant le corpus OPUS (Tiedemann, 2009) pour la paire de langues français-arabe. Ce corpus regroupe 74 067 paires de phrases parallèles extraites des résolutions des

Nations Unies. Ces résolutions citent certains noms de dirigeants, et beaucoup de noms de pays et d'organisations. Nous avons divisé ce corpus en trois parties : 70 067 paires de phrases pour l'apprentissage du modèle de traduction, 3 500 paires de phrases pour la construction du lexique bilingue en utilisant notre aligneur de mots et 500 paires de phrases pour l'évaluation du système de traduction Moses. Pour estimer le modèle de traduction du système de référence, nous avons construit un corpus d'apprentissage contenant 70 067 paires de phrases auquel nous avons ajouté les 3 500 paires de phrases utilisées pour l'alignement de mots. Pour étudier l'impact du lexique bilingue produit par l'outil d'alignement de mots intégrant la translittération sur le modèle de traduction du système Moses, nous avons ajouté ce lexique bilingue construit à partir des 3 500 paires de phrases au corpus d'apprentissage. Le modèle de traduction utilisé est appris sur les lemmes des mots composant le corpus parallèle d'apprentissage et les lemmes des mots produits par notre aligneur. Nous avons aussi entraîné un modèle de langue (tri-grammes) sur la totalité du corpus OPUS en langue arabe (74 067 phrases) en utilisant la boite à outils IRSTLM (Federico et al., 2008). Deux types de corpus de test ont été utilisés pour mener nos expérimentations : *Tout-Corpus-Test* et *Noms-propres-Corpus-Test*. Le premier corpus de test *Tout-Corpus-Test* est constitué de 500 paires de phrases parallèles extraites aléatoirement du corpus OPUS. Pour mesurer l'apport réel du lexique bilingue des noms propres translittérés, nous avons constitué un corpus de test noté *Noms-propres-Corpus-Test* où nous ne conservons que les phrases du corpus *Tout-Corpus-Test* contenant au moins un nom propre. Ce corpus contient 173 paires de phrases parallèles. La qualité de traduction du système de référence (celui qui n'intègre pas les translittérations) ainsi que celui intégrant les translittérations est évaluée sur les deux corpus de test sur la base de la métrique BLEU (Papineni et al., 2002). Nous avons préféré utiliser la métrique BLEU car elle est la plus appropriée pour évaluer les systèmes de traduction statistique à base de séquences (n-grammes) tels que Moses. Nous avons considéré qu'à chaque phrase source correspond une seule phrase de référence en langue cible. Les résultats de traduction obtenus pour les deux configurations sont regroupés dans la table 4.

Corpus d'apprentissage	Tout-Corpus-Test	Noms-propres-Corpus-Test
sans les résultats de l'appariement de cognats (sans translittération)	15,79	17,67
avec les résultats de l'appariement de cognats (avec translittération)	16,49	19.52

TABLE 4 : Résultats de traduction selon le score BLEU

Tout d'abord, nous constatons que le score BLEU obtenu est satisfaisant compte tenu de la taille du corpus d'apprentissage et du modèle de traduction utilisé et qui a été estimé sur des lemmes plutôt que sur des formes de surface (Sadat, Habash, 2006). Ce score varie en fonction du type du jeu de test. Le corpus de test *Noms-propres-Corpus-Test* qui ne considère que les phrases contenant des noms propres du lexique bilingue rapporte des scores BLEU plus élevés que le corpus de test *Tout-Corpus-Test* dans les deux configurations (corpus d'apprentissage sans l'ajout de translittération ou avec translittération). Les résultats obtenus montrent que l'intégration dans le corpus d'apprentissage du modèle de traduction des alignements obtenus par le module d'appariement de cognats utilisant la translittération a permis d'obtenir un gain de +0,70 points BLEU pour le corpus de test *Tout-Corpus-Test* et un gain de +1,85 pour le corpus de test *Noms-propres-Corpus-Test*. Ces résultats confirment ceux de (Huang et al., 2003) qui ont obtenu une F-Mesure de 81% pour l'alignement d'entités nommées à partir d'un corpus parallèle chinois-anglais et un gain de +0,06 en score NIST pour la traduction.

Pour évaluer la significativité statistique des résultats obtenus, nous utilisons la méthode par ré-échantillonnage par amorce décrite par (Koehn, 2004). Cette méthode estime la probabilité (p-valeur) qu'une différence mesurée entre les scores BLEU surgit par hasard et ce par la création à plusieurs reprises (10 fois) d'échantillons uniformes avec remise à partir des corpus de tests. Nous exploitons cette méthode pour comparer les deux configurations (corpus d'apprentissage sans l'ajout de translittération ou avec translittération) selon le corpus de test utilisé. Sur un intervalle de confiance (IC) de 95%, les résultats varient de non significatifs (quant $p > 0.05$) à hautement significatifs. Les p-valeurs obtenues sur les corpus de test *Tout-Corpus-Test* et *Noms-propres-Corpus-Test* sont respectivement de 0,02 et 0,01. Par conséquent, les améliorations apportées par l'utilisation de la translittération sont significatives dans les deux configurations de test.

5 Conclusion et travaux futurs

Nous avons décrit dans cet article, d'une part, un système de translittération des noms propres de l'écriture arabe vers l'écriture latine, et d'autre part, un outil d'alignement de mots simples et composés à partir de corpus de textes parallèles français-arabe. Nous nous sommes particulièrement intéressés à l'étude de l'impact de l'utilisation de la translittération sur la qualité du lexique bilingue produit par l'outil d'alignement de mots. Pour réaliser cette étude, nous avons évalué l'outil d'alignement de mots intégrant la translittération en utilisant deux approches : une évaluation de la qualité

d'alignement à l'aide d'un alignement de référence construit manuellement et une évaluation de l'impact de cet alignement sur la qualité de traduction du système de traduction automatique statistique Moses. Les résultats obtenus montrent que la translittération améliore aussi bien la qualité de l'alignement de mots que celle de la traduction. Dans nos expérimentations sur l'outil d'alignement de mots, le modèle de traduction a été estimé sur des lemmes plutôt que sur des formes de surface qui généralement diminue la qualité de traduction plus particulièrement pour une langue morphologiquement riche comme l'arabe. De même, les traductions du lexique bilingue produit par l'outil d'alignement de mots ne sont pas pondérées, ce qui nous prive d'intégrer ce lexique directement dans la table de traduction. Nos travaux futurs sur l'alignement de mots s'orientent, d'une part, vers l'utilisation d'un modèle de génération pour produire les formes de surface adéquates à partir des résultats de traduction présentés en lemmes dans cette étude, et d'autre part, vers une amélioration des résultats de notre outil d'alignement en lui intégrant l'appariement d'expressions multimots et en pondérant les traductions du lexique bilingue qu'il produit. Par ailleurs, nos expérimentations sur le système de translittération ont montré que les corpus étudiés contenaient aussi des noms propres latins et que la précision de l'alignement de mots est très élevée lorsque des noms propres arabes sont présents dans les phrases source et cible. Nos travaux futurs en translittération s'orientent vers une prise en compte plus large des noms propres latins.

Références

ABDULHAY A. (2012). Constitution d'une ressource sémantique arabe à partir d'un corpus multilingue aligné. *Thèse de Doctorat de l'Université Stendhal – Grenoble III.*

ABDULJALEEL N., LARKEY L. (2003). Statistical transliteration for English-Arabic Cross Language Information Retrieval. *Proceedings of the Twelfth ACM International Conference on Information and Knowledge Management*, New Orleans, Louisiana, 139-146.

ALGHAMDI M. (2005). Alghorithms for Romanizing Arabic names. *Journal of King Saud University - Computer and Information Sciences*,Volume 17, Riyadh, 105-128.

ALLAUZEN A., WISNIEWSKI G. (2009). Modèles discriminants pour l'alignement mot à mot. *TAL Volume 50 – n° 3/2009,* 173 – 203.

AL-ONAIZAN Y., KNIGHT K. (2002). Translating named entities using monolingual and bilingual resources. *Proceedings of the 40th ACL Conference*, USA.

BARKAT-DEFRADAS M., HAMDI R., PELLEGRINO F. (2004). De la caractérisation linguistique à l'identification automatique des dialectes arabes. *Proceedings of MIDL 2004,* 51-56.

BEESLEY K. R., KARTTUNEN L. (2003). Finite State Morphology. *Stanford, CA: CSLI Publications*.

BESANÇON R., DE CHALENDAR G., FERRET O., GARA F., LAIB M., MESNARD O., SEMMAR N. (2010). LIMA: A Multilingual Framework of Linguistic Analysis and Linguistic Resources Development and Evaluation. *Proceedings of LREC 2010,* 3697-3704.

BOUAMOR D., SEMMAR N., ZWEIGENBAUM P. (2012). Identifying bilingual Multi-Word Expressions for Statistical Machine. *Proceedings of the Eighth international conference on Language Resources and Evaluation (LREC), Turkey.*

BROWN P. F., PIETRA S. A. D., PIETRA V. J. D., MERCER R. L. (1993). The mathematics of statistical machine translation : parameter estimation. *Computational Linguistics*, *Volume 19, Number 2,* 263-311.

DAILLE B., GAUSSIER E., LANGE J.-M. (1994). Towards automatic extraction of monolingual and bilingual terminology. *Proceedings of the 15th International Conference on Computational Linguistics (COLING'94)*, 515-521.

DEBILI F., ZRIBI A. (1996). Les dépendances syntaxiques au service de l'appariement des mots. *Actes du 10ème Congrès Reconnaissance des Formes et Intelligence Artificielle (RFIA'96)*.

FEDERICO M., BERTOLDI N., CETTOLO M. (2008). IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models. *Proceedings of Interspeech, Australia, 2008.*

FRUNZA O., INKPEN D. (2009). Identification and Disambiguation of Cognates, False Friends, and Partial Cognates Using Machine Learning Techniques. *International Journal of Linguistics, Vol. 1*.

GAUSSIER E., LANGE J. M. (1995). Modèles statistiques pour l'extraction de lexiques bilingues. *Traitement Automatique des Langues, Volume 36. ATALA,* 133-155.

GERMANN U. (2008). Yawat: Yet Another Word Alignment Tool. *Proceedings of ACL 2008, Columbus,* 20-23

GUIDERE M. (2004). Le traitement de la parole et la détection des dialectes arabes. *Langues stratégiques et défense nationale, Publications du CREC, Saint-Cyr,* 53–75.

HUANG F., VOGEL S., WAIBEL A. (2004). Improving named entity translation combining phonetic and semantic similarities. *Proceedings of HLT-NAACL 2004,* 281-288.

HUANG F., VOGEL S., WAIBEL A. (2003). Automatic Extraction of Named Entity Translingual Equivalence Based on Multi-feature Cost Minimization. *Proceedings of the 41st Annual Conference of the Association for Computational Linguistics (ACL'03), Workshop on Multilingual and Mixed-language Named Entity Recognition, Sapporo, Japan.*

HULDEN M. (2009). Foma: a Finite-State Compiler and Library. *Proceedings of: EACL 2009, 12th Conference of the European Chapter of the Association for Computational Linguistics,* Athens, Greece, 29-32.

JARO M. A. (1989). Advances in record linkage methodology as applied to the 1985 census of Tampa Florida. *Journal of the American Statistical Association 84*, 414–420.

JIANG L., ZHOU M., CHIEN L. F., NIU C. (2007). Named entity translation with web mining and transliteration. *Proceedings of the 20th International Joint Conference on Artif cial Intelligence*, 1629-1634.

KOEHN P., HOANG H., BIRCH A., CALLISON-BURCH C., FEDERICO M., BERTOLDI N., COWAN B., SHEN W., MORGAN C., ZENS R., DYER C., BOJAR O., CONSTANTIN A., HERBST E. (2007). Moses: Open source toolkit for statistical machine translation. *Proceedings of ACL* 2007, 177-180.

KOEHN P. (2004). Statistical significance tests for machine translation evaluation. *Proceedings of EMNLP 2004.*

KNIGHT K., GRAEHL J. (1997). Machine transliteration. *Journal version Computational linguistics,* 24(4), 599-612.

KONDRAK G. (2005). Cognates and Word Alignment in Bitexts. *Proceedings of the Tenth Machine Translation Summit (MT Summit X), Thailand*.

KONDRAK G., MARCU D., KNIGHT K. (2003). Cognates Can Improve Statistical Translation Models. *Proceedings of HLT-NAACL 2003*, 46-48.

KRAIF O. (2001). Exploitation des cognats dans les systèmes d'alignement bi-textuel: architecture et évaluation. *TAL, 42(3),* 833-867.

LEFEVER E., MACKEN L., HOSTE V. (2009). Language-independent bilingual terminology extraction from a multilingual parallel corpus. *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Greece*.

MIHALCEA R., PEDERSEN T. (2003). An evaluation exercise for word alignment. *Proceedings of The HLT-NAACL 2003 Workshop on Building and using parallel texts: data driven machine translation and beyond*, 1-10.

OZDOWSKA S., CLAVEAU V. (2006), Inférence de règles de propagation syntaxique pour l'alignement de mots. *TAL, Volume 47, n°1 ATALA,* 167-186.

OCH F. J., NEY H. (2000). Improved Statistical Alignment Models. *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics,* 440-447.

OKITA T., GUERRA M., ALFREDO GRAHAM Y., WAY A. (2010). Multi-word expression sensitive word alignment. *Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, 26–34*.

PAPINENI K., ROUKOS S., WARD T., ZHU W. J. (2002). Bleu: a method for automatic evaluation of machine translation. *Proceedings of the 40th Annual meeting of the Association for Computational Linguistics,* 311–318.

RILEY M., ALLAUZEN C., MARTIN J. (2009). OpenFst: An Open-Source, Weighted Finite-State Transducer Library and its Applications to Speech and Language. *Proceedings of NAACL HLT 2009: Tutorials,* 9–10.

POULIQUEN B., STEINBERGER R. (2007). Acquisition and Use of Multilingual Name Dictionaries. *Proceedings of the Workshop Acquisition and Management of Multilingual Lexicons (AMML'2007) - RANLP'2007, Bulgaria.*

SAÂDANE H., SEMMAR N. (2012). Utilisation de la translittération arabe pour l'amélioration de l'alignement de mots à partir de corpus parallèles français-arabe. *Actes TALN 2012*, 127-140.

SADAT F., HABASH N. (2006). Combination of Arabic Preprocessing Schemes for Statistical Machine Translation. *Proceedings of ACL 2006,* 1-8.

SEMMAR N., SERVAN C., DE CHALENDAR G., LE NY B. (2010). A Hybrid Word Alignment Approach to Improve Translation Lexicons with Compound Words and Idiomatic Expressions. *Proceedings of the 32nd Translating and the Computer conference, England*.

SERETAN V., WEHRLI E. (2007). Collocation translation based on sentence alignment and parsing. *Actes de TALN 2007*.

SHAO L., NG H. T. (2004). Mining new word translations from comparable corpora. *Proceedings of the 20th International Conference on Computational Linguistics (COLING'04),* 618-624.

SHERIF T., KONDRAK G. (2007). Bootstrapping a stochastic transducer for Arabic-English transliteration extraction. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007),* 864-871.

SIMARD M., FOSTER G. F., ISABELLE P. (1993). Using cognates to align sentences in bilingual corpora. *Proceedings of the Conference of the Centre for Advanced Studies on Collaborative Research,* 1071-1082.

STALLS B., KNIGHT K. (1998). Translating names and technical terms in Arabic text. *Proceedings of the COLING/ACL Workshop on Computational Approaches to Semitic Languages,* Montreal, Québec, 34-41.

TAO T., YOON S. Y., FISTER A., SPROAT R., ZHAI C. (2006). Unsupervised named entity transliteration using temporal and phonetic correlation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'06),* 250-257.

TIEDEMANN J. (2009). News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. *N. Nicolov and K. Bontcheva and G. Angelova and R. Mitkov (eds.) Recent Advances in Natural Language Processing, Volume V,* 237-248.

TUFIS I., ION R. (2007). Parallel corpora, alignment technologies and further prospects in multilingual resources and technology infrastructure. *Proceedings of the 4th International Conference on Speech and Dialogue Systems,* 183–195.

VERONIS J., HAMON O., AYACHE C., BELMOUHOUB R., KRAIF O., LAURENT D., NGUYEN T. M. H., SEMMAR N., STUCK F., ZAGHOUANI W. (2008). Arcade II Action de recherche concertée sur l'alignement de documents et son évaluation. *Chapitre 2, Editions Hermès.*

VINTAR S., FISIER D. (2008). Harvesting multi-word expressions from parallel corpora. *Proceedings of LREC, Morocco.*

WINKLER W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. *Section on Survey Research Methods, American Statistical Association,* 354–359.

YASER A. O., KNIGHT K. (2002). Translating named entities using monolingual and bilingual resources. *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics (ACL'02),* 400-408.

Adaptation thématique pour la traduction automatique de dépêches de presse

Souhir Gahbiche-Braham[1, 2] Hélène Bonneau-Maynard[1, 2] François Yvon[1]
(1) LIMSI-CNRS, B.P. 133, F-91403 Orsay Cedex, France
(2) Université Paris Sud
souhir@limsi.fr, hbm@limsi.fr, yvon@limsi.fr

Résumé. L'utilisation de méthodes statistiques en traduction automatique (TA) implique l'exploitation de gros corpus parallèles représentatifs de la tâche de traduction visée. La relative rareté de ces ressources fait que la question de l'adaptation au domaine est une problématique centrale en TA. Dans cet article, une étude portant sur l'adaptation thématique des données journalistiques issues d'une même source est proposée. Dans notre approche, chaque phrase d'un document est traduite avec le système de traduction approprié (c.-à-d. spécifique au thème dominant dans la phrase). Deux scénarios de traduction sont étudiés : (a) une classification *manuelle*, reposant sur la codification IPTC ; (b) une classification *automatique*. Nos expériences montrent que le scénario (b) conduit à des meilleures performances (à l'aune des métriques automatiques), que le scénario (a). L'approche la meilleure pour la métrique BLEU semble toutefois consister à ne pas réaliser d'adaptation ; on observe toutefois qu'adapter permet de lever certaines ambiguïtés sémantiques.

Abstract. Statistical approaches used in machine translation (MT) require the availability of large parallel corpora for the task at hand. The relative scarcity of thes resources makes domain adaptation a central issue in MT. In this paper, a study of thematic adaptation for News texts is presented. All data are produced by the same source : News articles. In our approach, each sentence is translated with the appropriate translation system (specific to the dominant theme for the sentence). Two machine translation scenarios are considered : (a) a *manual* classification, based on IPTC codification ; (b) an *automatic* classification. Our experiments show that scenario (b) leads to better performance (in terms of automatic metrics) than scenario (a) . The best approach for the BLEU metric however seems to dispense with adaptation alltogether. Nonetheless, we observe that domain adaptation sometimes resolves some semantic ambiguities .

Mots-clés : adaptation thématique, classification automatique, traduction automatique.

Keywords: domain adaptation, automatic classification, machine translation.

1 Introduction

En traitement automatique des langues, l'adaptation au domaine[1] est une question souvent abordée. Pour l'application de traduction automatique, adapter ou spécialiser des modèles de traduction à un genre ou à un thème peut permettre de résoudre certaines ambiguïtés qui ne pourraient être levées par des modèles généraux. Ainsi, si le domaine considéré est celui de la réservation de billets d'avions, on peut s'attendre à ce que la spécialisation de modèles de traduction à ce domaine permette de désambiguïser du mot anglais *book*, qui peut être traduit en français par le substantif *livre* ou par le verbe *réserver*. Conjointement, l'adaptation du modèle de langue permet de distinguer certaines traductions en utilisant le contexte de la phrase en langue cible. Sennrich (2012b) explore les différences conceptuelles entre l'adaptation des modèles de traduction et celle du modèle de langue ainsi que leurs effets sur la performance de la traduction.

Dans l'étude présentée ici, la question de l'adaptation est abordée à un niveau plus fin que celui qui est classiquement considéré : alors qu'on se pose le plus souvent la question de combiner des données *du domaine* avec des données *hors domaine*, on envisage ici une adaptation **thématique** au sein d'un même registre et genre, celui des dépêches journalistiques. Pour ces documents, une classification manuelle en thèmes (ou catégories) est attribuée par les journalistes lors de la rédaction des dépêches, que nous cherchons donc à utiliser. Notre corpus est en effet constitué d'un ensemble de

[1]Le terme « domaine » doit être compris dans une acception assez vague : le besoin d'adapter se révèle dès lors que les données d'apprentissage diffèrent des données de l'application, que ces différences soient des différences de genre, de registre, de modalité, ou encore de thème, qui est la situation considérée ici.

dépêches produites par l'AFP[2]. Ces documents sont classifiés selon les 17 catégories principales du standard IPTC[3] (voir section 3). L'utilisation de cette classification est une des particularités de notre étude. La deuxième particularité, qui est inhérente au fait que les données proviennent d'une même source, est qu'elles sont beaucoup plus homogènes que dans les cas standard d'adaptation au domaine. Par ailleurs, plusieurs catégories peuvent être affectées à une même dépêche. De plus, les observations sur le corpus montrent que les frontières entre les catégories sont relativement floues.

Dans ce contexte, différentes méthodes d'adaptation thématique sont explorées, qui impliquent aussi bien l'adaptation des modèles de traduction que l'adaptation des modèles de langue. Les premières expériences sont menées en utilisant la classification *manuelle* en catégories qui accompagne les dépêches ; dans ce, cas toutes les phrases d'une même dépêche sont traduites avec les mêmes modèles. Nous proposons ensuite de considérer une classification *automatique* des phrases, permettant de traduire chaque phrase par le modèle qui lui est le plus approprié.

Cet article est organisé comme suit : la section 2 présente un état de l'art des approches d'adaptation au domaine en traduction automatique. Les données utilisées pour cette étude sont décrites dans la section 3. La section 4 décrit notre approche et la section 5 expose les résultats obtenus. Les principales conclusions sont enfin présentées à la section 6.

2 État de l'art

Les premiers travaux sur l'adaptation des modèles de langue ont été publiés durant les années 90, particulièrement dans le domaine de la reconnaissance de la parole (De Mori & Federico, 1999). La relative rareté des ressources parallèles qui sont nécessaires à l'apprentissage de systèmes de traduction a progressivement fait émerger cette problématique en TA, et a donné lieu dans les années récentes à une littérature abondante que nous survolons rapidement ici.

Langlais (2002) présente les premiers travaux sur l'adaptation du domaine en traduction automatique. Il implémente une stratégie qui consiste à compléter le modèle de traduction avec lexiques adaptés. De nombreuses stratégies d'adaptation ont ensuite été proposées, pour l'essentiel fondées sur l'interpolation de modèles du domaine et de modèles hors-domaine, comme par exemple dans (Koehn & Schroeder, 2007), où les meilleures performances sont obtenues par une interpolation linéaire de modèles de langue et une interpolation log-linéaire de modèles de traduction.

Des approches d'adaptation fondées sur les modèles de mélange ont été également été proposées par Foster & Kuhn (2007) et par Sennrich (2012a). Ces auteurs utilisent des modèles de langue et des modèles de traduction adaptés combinés par interpolation linéaire et log-linéaire. Sennrich (2012b) adapte les modèles de traduction en minimisant les scores de perplexité et pour optimiser les coefficients d'interpolation. Si l'adaptation thématique améliore souvent les performances, il apparaît également que les données hors-domaine peuvent dégrader les performances de traduction en introduisant des ambiguïtés lexicales. En particulier, Haddow & Koehn (2012) montrent que l'ajout de données hors domaine à un corpus d'apprentissage peut améliorer la traduction des mots rares (les moins fréquents) mais en revanche dégrade la qualité de la traduction des mots les plus fréquents.

Une stratégie alternative d'adaptation consiste à pondérer différentiellement les données du domaine et hors-domaine ; cette piste est explorée notamment par Foster *et al.* (2010), Shah *et al.* (2010) ou encore Niehues & Waibel (2010) : des poids sont assignés aux phrases et aux segments de phrases avant la création des modèles de traduction. Il est par exemple possible de nuancer l'importance des phrases hors-domaine sur la base d'un degré de similarité avec des phrases du domaine.

Selon les applications, il est possible de considérer des distinctions plus fines que *in domain* versus *out-of-domain*, Dans ce contexte, Yamamoto & Sumita (2008) détectent les domaines de chaque phrase à traduire, qui est ensuite traduite en utilisant les modèles spécifiques au domaine détecté. Nakov (2008) utilise une interpolation log-linéaire et combine plusieurs modèles de langue de différents domaines. De nouveaux traits sont ajoutés, un pour chaque modèle de traduction interpolé. Ce trait indique pour chaque paire de phrases, si elle provient du modèle de traduction en question.

Zhao *et al.* (2004) explorent des techniques pour l'adaptation non-supervisée des modèles de langue. Ces modèles sont construits à partir de données monolingues extraites en se basant sur des scores de similarité et des traits (*features*) sémantiques. Ces modèles sont interpolés avec un modèle de langue général contenant l'ensemble des données et permettent d'obtenir une amélioration de la qualité de traduction. Dans des travaux plus récents, Sennrich *et al.* (2013) proposent une approche d'adaptation non supervisée dans laquelle plusieurs modèles de langue sont interpolés log-linéairement lors de

[2]Agence France Presse, `http://www.afp.fr`.

[3]L'IPTC – *International Press Telecommunications Council* – est un consortium qui rassemble les grandes agences de presse mondiales. L'IPTC fournit des schémas de classification, de normalisation et de codage des métadonnées, voir `http://www.iptc.org`.

la traduction. Des scores sont calculés lors du décodage pour choisir la meilleure hypothèse de traduction pour chaque phrase source.

Dans la plupart des travaux existants, les données adaptées proviennent de sources bien séparées. Un exemple extrême est donné par Sennrich (2012b), qui combine des données extraites à partir du journal *Alpine Club* (dédié à l'alpinisme) et des données issues du corpus *Europarl* (débats politiques). Il est clair que ces données dérivent de sources bien séparées. La situation est un peu moins marquée dans le travail de Banerjee (2012) sur la traduction de documents techniques, qui utilise deux catégories de données : ceux qui traitent de logiciels du domaine *availability* (récupération de données, sauvegarde) et ceux qui traitent du domaine *security* (vulnérabilité de logiciels malveillants, protection contre les attaques).

Eidelman *et al.* (2012) proposent une approche de classification thématique non-supervisée fondée sur les modèles d'allocation de Dirichlet latente (LDA). Le thème de chaque phrase est induit de manière non supervisée : des distributions thématiques sont utilisées pour calculer les probabilités de pondération lexicales du thème dépendant et les intégrer dans le modèle de traduction comme étant des traits (*features*). Les données utilisées proviennent du corpus FBIS[4] (audio), et du corpus NIST pour la traduction du chinois vers l'anglais.

Dans cet article, nous proposons plusieurs approches pour adapter les modèles de traduction et les modèles de langue. Nous nous intéressons à deux approches de classification (*manuelle* et *automatique*), qui servent à faire en sorte que chaque phrase soit traduite avec le système approprié construit à partir de modèles adaptés. Notre approche est semblable à celle de Sennrich (2012a) et de celle de de Yamamoto & Sumita (2008). La différence est que dans notre cas, les données proviennent de la même source et que l'on dispose d'une catégorisation manuelle des documents.

3 Description des données

Les données utilisées dans cette étude sont constituées d'un ensemble de dépêches journalistiques en arabe et en français produites par l'AFP entre décembre 2009 et juillet 2012. Chaque dépêche est catégorisée manuellement par les journalistes en utilisant les 17 catégories principales de la nomenclature IPTC (voir tableau 1). Une dépêche peut être affectée à plusieurs catégories. Il peut arriver également qu'une dépêche ne soit pas catégorisée par la suite d'un oubli.

Le corpus parallèle utilisé pour construire un système de traduction automatique statique est constitué de 265 000 paires de phrases extraites d'un corpus comparable avec la méthode décrite dans (Gahbiche-Braham *et al.*, 2011). Le tableau 1 donne la répartition des phrases selon ces 17 catégories. Pour chaque catégorie et pour chaque langue, la valeur indiquée est le pourcentage des phrases issues de dépêches étiquetées avec cette catégorie. Une phrase pouvant appartenir à une ou plusieurs catégories, on notera que la somme de ces pourcentages est supérieure à 100 %.

On peut observer que les valeurs diffèrent pour l'arabe et le français. En effet, bien que les dépêches en arabe soient souvent des (quasi)-traductions des dépêches en français, la catégorie thématique des dépêches est réattribuée manuellement par les journalistes après la traduction. Notons que la distribution des données est très inégale ; la catégorie dominante est la catégorie POL avec environ 60 % des phrases appartenant à cette catégorie. Nous limiterons notre étude aux trois catégories les plus fréquentes : *politique* (POL), *guerre* (WAR), et *finance* (FIN), les autres catégories étant présentes en trop petit nombre pour pouvoir mener des études. Le tableau présente également la répartition des phrases parallèles en catégories pour les corpus d'entraînement, de développement et de test initial. Le tableau 2 met en évidence le nombre de phrases communes pour chaque paire des catégories POL, WAR et FIN. On observe notamment que les catégories POL et WAR ont une intersection importante, représentant plus de 16 % de l'ensemble des phrases des deux catégories, et que la proportion des phrases de la catégorie WAR qui sont également étiquetées par la catégorie POL atteint presque la moitié (42,7%). Bien que non négligeables, les intersections sont moins importantes pour les deux autres paires de catégories, avec cependant encore 40,3 % des phrases de la catégorie FIN également affectées à la catégorie POL.

Les données collectées pendant le mois de novembre 2011 ont été isolées afin de construire un ensemble de test initial et un ensemble de développement. Ces ensembles sont constitués de respectivement 1 000 et 1 178 phrases. Trois corpus de test et de développement spécifiques à chaque catégorie ont été également constitués. Ces derniers sont également constitués de 1 000 (test) et 1 178 (développement) phrases chacun.

[4]FBIS : Foreign Broadcast Information Service.

				# phrases parallèles		
Catégorie	% AR	% FR	Nom de la catégorie	Entraînement	dev	test
ACE	1,1	2,7	*arts, culture and entertainment*	2 825	3	1
CLJ	9,2	12,1	*crime, law and justice*	24 330	25	32
DIS	5,3	6,8	*disaster and accident*	13 951	19	10
FIN	**14,8**	**15,3**	*economy, business and finance*	39 227	132	162
EDU	0,2	0,2	*education*	420	0	0
EVN	0,8	1,3	*environmental issue*	2 137	1	0
HTH	0,9	1,0	*health*	2 302	2	0
HUM	0,4	0,7	*human interest*	1 067	2	1
LAB	0,8	2,0	*labour*	1 986	2	3
LIF	0,03	0,03	*lifestyle and leisure*	93	0	0
POL	**58,9**	**62,3**	*politics*	156 352	663	561
REL	3,4	3,9	*religion and belief*	8 966	6	8
SCI	1,0	1,3	*science and technology*	2 746	2	2
SOI	2,0	2,6	*social issue*	5 367	2	3
SPO	1,3	1,3	*sport*	3 327	8	8
WAR	**38,3**	**42,9**	*unrest, conflicts and war*	101 655	310	207
WEA	0,9	0,1	*weather*	2 393	1	2
-	0,1	2,3	-	-	-	-

TAB. 1 – Distribution des catégories IPTC en arabe et en français et nombre de phrases pour les corpus d'entraînement, de développement et de test initial.

			Pourcentage des phrases communes dans chaque cat.	
Catégorie 1	Catégorie 2	Phrases en commun (%)		
POL	WAR	16,35	27,7 % POL	42,7 % WAR
POL	FIN	5,97	12,5 % POL	40,3 % FIN
WAR	FIN	1,36	3,6 % WAR	9,2 % FIN

TAB. 2 – Pourcentage des phrases communes entre les paires de catégories POL-WAR, POL-FIN et WAR-FIN pour le corpus d'entraînement, et pourcentage des phrases communes existant dans chaque catégorie pour ces trois paires.

4 Adaptation thématique

L'idée principale consiste à utiliser une classification thématique du corpus d'entraînement pour produire des systèmes de traduction spécifiques aux différentes thématiques. Il s'agit d'étudier l'impact de l'utilisation de modèles de traduction et de modèles de langue spécifiques sur la qualité de la traduction, en particulier dans le but de lever certaines ambiguïtés de mots polysémiques. Par exemple le verbe سجل (enregistre) en arabe peut avoir deux sens *marquer* ou *enregistrer* qui sont traduits différemment en français. La phrase سجل فرحان شكور هدف العراق (*Farhan Chakour* ***a marqué*** *le but de l'Iraq*) est extraite d'une dépêche à laquelle la catégorie SPO (sport) est attribuée, alors que la phrase سجل سنودن ومساعدته اسميهما على رحلة ايرفلوت (*Snowden et son assistante* ***ont enregistré*** *leurs noms dans le vol Airfloat*) est extraite d'une dépêche à laquelle la catégorie POL (politics) lui est affectée. La connaissance de la catégorie de la phrase peut permettre de choisir entre les deux sens du mot سجل.

Les difficultés de cette étude viennent du fait que nous nous intéressons au cas où les données spécialisées dérivent d'une même source et sont telles que les frontières entre thématiques sont relativement floues. Les catégories que nous traitons ne sont pas exclusives et l'intersection entre deux catégories peut être importante.

Deux scénarios d'adaptation sont alors proposés : une adaptation utilisant *une classification manuelle* des phrases et une adaptation utilisant *une classification automatique* des phrases. La section 4.1 présente l'approche de classification automatique utilisée dans cet article. Les méthodes d'adaptation sont décrites dans la section 4.2.

4.1 Classification manuelle versus classification automatique

Dans une première approche, chaque phrase du corpus d'entraînement est classifiée *manuellement* selon la catégorie de la dépêche originale dont elle est extraite (catégorisée par l'AFP). Seulement 46 % des phrases d'entraînement appartiennent à une seule catégorie. Les phrases appartenant à plusieurs catégories sont affectées à chacune de ces catégories.

L'approche par *classification automatique* remet en question l'hypothèse précédente. Plutôt que de projeter systématiquement la (ou les) catégorie(s) d'une dépêche sur les phrases la constituant, chaque phrase est catégorisée indépendamment des autres phrases du document. La figure 1 justifie cette seconde approche. Les deux phrases sont extraites d'une dépêche étiquetée par la catégorie FIN. Si la catégorie FIN est appropriée pour la première phrase de cet extrait, cela est moins clair pour la seconde, qui n'a en fait rien de spécifique à la catégorie finance.

> La jeune entreprise TimoCom est rapidement devenue une entreprise de taille moyenne, [...]
> Le texte du communiqué issu d'une traduction ne doit d'aucune manière être considéré comme officiel.

FIG. 1 – Un extrait d'une dépêche AFP affectée à la catégorie FIN.

Aux trois catégories traitées dans cette étude, une catégorie générique *Autre* est ajoutée, qui permet comme expliqué ci-dessous de catégoriser toutes les phrases. Un classifieur est donc entraîné pour affecter automatiquement une catégorie parmis (POL, WAR, FIN ou Autre) à chaque phrase.

Ce classifieur de phrases (en arabe) implémente une version simplifiée de l'algorithme Espérance-Maximisation (EM) pour le modèle de mélange de lois multinomiales (Rigouste *et al.*, 2007), initialisé avec les catégories IPTC. La principale singularité de l'approche consiste à estimer également un modèle « généraliste » en plus des modèles spécialisés. Lorsqu'une phrase est trop courte, ou bien qu'aucune des trois catégories n'obtient une vraisemblance meilleure que le modèle généraliste, alors la phrase n'est affectée à aucune des trois catégories et sera traduite (au test) par un modèle généraliste agrégeant toutes les données disponibles. Cette stratégie a pour effet de spécialiser les catégories automatiques, mais également de réduire les données utilisées pour apprendre les modèles spécialisées.

La probabilité *a posteriori* de chaque phrase est calculée par rapport aux quatre modèles disponibles. La catégorie assignée à chaque phrase est celle du modèle pour lequel elle présente la probabilité *a posteriori* la plus grande. De nouveaux modèles spécifiques sont alors appris sur la base de cette nouvelle annotation, puis utilisés pour reclassifier les données jusqu'à la convergence comme schématisé dans la figure 2.

La figure 3 donne la répartition du nombre de phrases pour chaque catégorie. Le nombre de phrases pour les catégories spécifiques est beaucoup plus réduit avec une classification automatique qu'avec la classification manuelle car dans le premier cas on impose la contrainte qu'une phrase ne peut être affectée qu'à une seule catégorie.

Après apprentissage, toutes les catégories des phrases d'entraînement sont recalculées. De nouveaux modèles spécifiques constitués sur la base de cette classification automatique sont alors construits : un effet induit important est que les phrases « génériques »sont retirées du corpus d'apprentissage des modèles thématiques initiaux (construits par classification manuelle), ce qui rend les modèles thématiques plus spécialisés. On note également qu'avec la nouvelle classification, les phrases extraites d'une même dépêche peuvent être affectées à des catégories différentes.

4.2 Méthodes d'adaptation

Pour chacune des méthodes de classification considérées, quatre approches d'adaptation sont explorées :

(a) Fusion des modèles de traduction : des modèles de traduction spécifiques sont entraînés séparément pour chaque

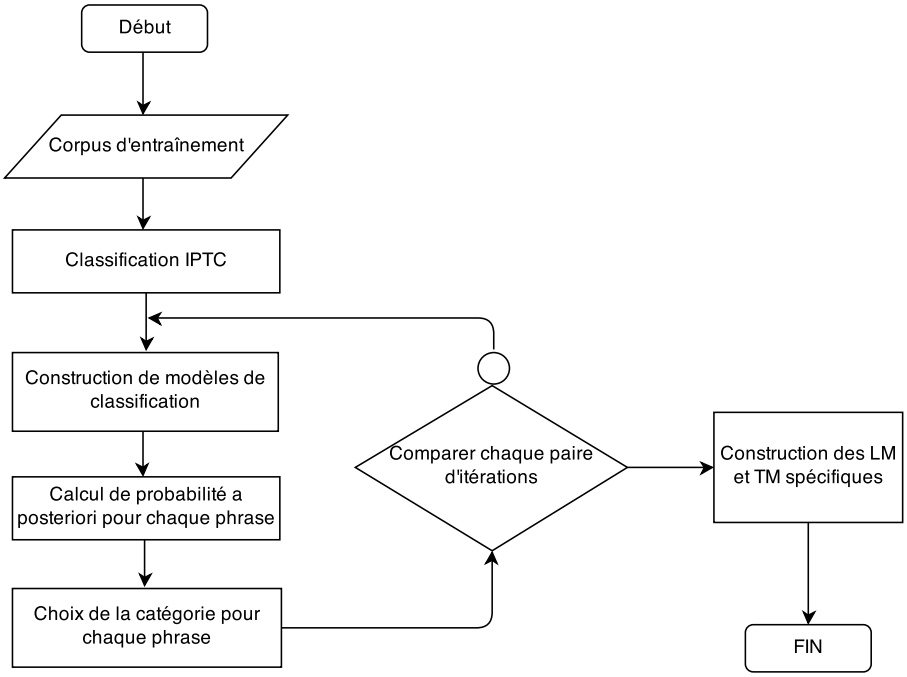


FIG. 2 – Processus itératif pour la construction du classifieur automatique.

catégorie (POL, WAR et FIN). Ils sont ensuite fusionnés en un seul modèle de traduction global. Chaque segment (chaque *phrase*) de la table de traduction globale (*Merge*), est ainsi nanti de 17 paramètres : les 4 paramètres classiques pour chaque modèle ($P(s|t)$, $P(t|s)$, $lex(s|t)$ et $lex(t|s)$) – qui sont soit recopiés des modèles spécifiques si le segment y est présent, soit affectés d'un score de probabilité faible – et un score qui représente la probabilité de distorsion (constante 2,718) ;

(b) Interpolation log-linéaire de modèles de traduction : consistant à employer deux modèles de traduction (spécifique et générique) en privilégiant le premier modèle de traduction spécifique par rapport au modèle de traduction générique (mode *either* dans Moses avec l'option *decoding-graph-backoff*). Leurs poids sont optimisés simultanément avec MERT (Och, 2003) afin d'optimiser les performances de traduction ;

(c) Interpolation log-linéaire de modèles de langue, consistant à utiliser deux modèles de langues (spécifique et générique) ;

(d) Interpolation linéaire des modèles de langue : chacun des modèles de langue est donc d'abord entrainé sur un corpus spécifique, puis les modèles sont interpolés linéairement en utilisant des coefficients d'interpolation estimés en minimisant la perplexité sur un corpus de développement.

Les modèles génériques *Gen* englobant toutes les données, sont construits dans le but d'avoir des modèles de traduction et de langue indépendant du domaine.

5 Expérimentations et résultats

Pour la traduction automatique, le décodeur à base de segments Moses[5] (Koehn *et al.*, 2007) est utilisé ; pour la phase d'entraînement, nous avons recours à l'aligneur sous-phrastique MGIZA++[6] (Gao & Vogel, 2008). La table de traduction est constituée en rendant symétriques les alignements selon l'heuristique *grow-diag-final-and* de Moses, et contient des segments dont la longueur va jusqu'à sept mots. L'outil SAPA (Gahbiche-Braham *et al.*, 2012) a été utilisé pour le prétraitement de l'arabe, consistant en particulier à normaliser les proclitiques et à présegmenter les tokens arabes en des unités plus courtes et plus facilement appariables avec des mots français. Le protocole expérimental est présenté dans

[5] http://www.statmt.org/moses/.
[6] http://www.kyloo.net/software/doku.php/mgiza:overview.

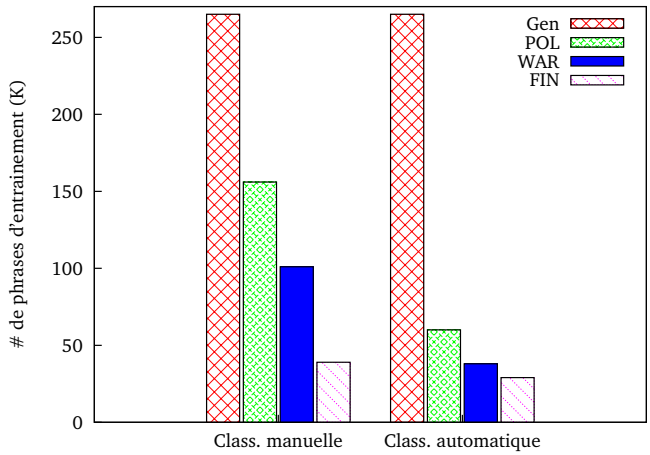


FIG. 3 – Nombre de phrases d'apprentissage par catégorie pour les deux scénarios : classification manuelle et automatique.

la section 5.1. Les modèles spécifiques sont évalués dans la section 5.2 et les différentes stratégies d'adaptation dans la section 5.3.

5.1 Protocole expérimental

Le corpus d'entraînement initial décrit dans la section 3 a été subdivisé pour construire des sous-corpus spécifiques à chacune des catégories IPTC choisies. Une *vérification manuelle du corpus de test initial* a été réalisée pour attribuer une seule catégorie à chaque phrase de l'ensemble de test initial (contenant initialement 40 % de phrases multicatégoriques). Pour les phrases multicatégoriques, la catégorie attribuée manuellement est une des catégories attribuées à la dépêche dont elle est extraite. Les phrases ne correspondant à aucune des catégories (POL, WAR et FIN) considérées dans l'étude sont affectées à la catégorie *Autre*. Le tableau 3 permet de comparer la répartition des phrases du corpus de test initial selon chaque catégorie, dans le cas d'un étiquetage manuel en comparaison à un étiquetage automatique. Avec la classification

	# phrases (%)	
Catégorie	Class. manuelle	Class. automatique
POL	56,1	33,7
WAR	20,7	15,8
FIN	16,2	14,9
Autre	7	35,6

TAB. 3 – Pourcentage des phrases pour chaque catégorie du corpus de test initial par rapport à l'ensemble des phrases, vérifié manuellement et classifié automatiquement.

automatique, 35,6 % des phrases sont attribuées à d'autres catégories que les trois catégories spécifiques.

Le tableau 4 montre le pourcentage des phrases ayant gardé les mêmes catégories pour la classification manuelle et automatique ainsi que le pourcentage des phrases ayant été classifiées dans les autres catégories. On observe que la classification automatique attribue la même catégorie que celle de l'étiquetage manuel dans environ 44 % des cas (la somme des termes sur la diagonale). Pour la catégorie POL par exemple, 25,3 % des phrases parmi 56,1 % ont gardé la même catégorie (c'est-à-dire 45 % des phrases étiquetées initialement POL). Avec la classification automatique, 33,7 % des phrases du corpus de test initial sont affectées à la catégorie POL (voir tableau 3), parmi ces phrases 5 % ont été initialement attribuées à la catégorie WAR, 2,2 % à la catégorie FIN et 1,2 % à d'autres catégories (première colonne du tableau 4).

On observe que pour certaines phrases, la nouvelle catégorie assignée automatiquement est plus appropriée que la catégorie assignée manuellement. C'est par exemple le cas de la phrase « *Afghanistan : six civils tués par une bombe artisanale*

Man \ Auto	POL	WAR	FIN	Autre
POL	**25,3**	6,1	5,5	19,2
WAR	5	**6,9**	0,9	7,9
FIN	2,2	1,2	**7,8**	5
Autre	1,2	1,6	0,7	**3,5**

TAB. 4 – Pourcentage des phrases ayant gardé les mêmes catégories pour la classification manuelle et automatique (en diagonale), ainsi que le pourcentage des phrases ayant été classifiées dans les autres catégories.

dans l'est. » initialement attribuée à la catégorie POL et reclassifiée dans la catégorie WAR ; ou encore de la phrase « *La zone euro sous pression pour maîtriser l'incendie de la dette.* » initialement attribuée à la catégorie WAR et reclassifiée automatiquement dans la catégorie FIN.

La traduction automatique de ce corpus de test initial, en utilisant l'ensemble des données d'entraînement, sans effectuer aucune adaptation aux catégories, donne un score BLEU de $33,47$.

5.2 Évaluation des modèles spécifiques

En se basant sur la classification de l'AFP, des modèles de langue (LM) spécifiques et des modèles de traduction spécifiques (TM) ont été construits. Un système de traduction général et trois systèmes de traduction spécifiques (POL, WAR et FIN) ont été construits. Le tableau 5 montre les résultats de la traduction automatique des trois tests spécifiques ainsi que le test initial (décrit dans le tableau 3) sur ces trois modèles spécifiques.

	BLEU			
Test	POL	WAR	FIN	Initial
POL	**30,94**	32,82	**29,89**	**31,73**
WAR	28,10	**32,96**	25,61	29,72
FIN	25,25	26,67	28,00	25,34

TAB. 5 – Évaluation des systèmes spécifiques – optimisés sur les corpus de développement spécifiques – sur les corpus de tests spécifiques POL, WAR et FIN et sur le test initial (présenté dans le tableau 3).

On observe que les tests spécifiques POL et WAR sont mieux traduits par les modèles spécifiques correspondants, ce qui confirme l'intérêt d'une traduction dépendant de la catégorie. En revanche, ceci n'est pas vrai pour le test spécifique FIN, pour lequel le meilleur résultat est obtenu avec le modèle spécifique POL. Ceci peut s'expliquer d'une part par le fait que le modèle spécifique FIN est entraîné sur quatre fois moins de données que le modèle POL, et d'autre part par le fait que 40 % des données utilisées pour entraîner le modèle POL sont aussi utilisées pour entraîner le modèle FIN (voir tableau 2). Le système spécifique POL donne également le meilleur résultat de traduction sur le corpus de test initial, puisque la plupart des phrases de ce test sont affectées à la catégorie POL dominante (voir tableau 3) et les modèles POL sont entraînés sur plus de données que pour les modèles WAR et FIN (voir figure 3).

5.3 Évaluation des stratégies d'adaptation

Les performances en BLEU des différentes méthodes d'adaptation décrites en section 4.2 sont reportées dans le tableau 6. Le tableau présente les résultats sur les tests spécifiques (classifiés manuellement) et sur le test initial (classifié manuellement et automatiquement).

Les résultats du système sans adaptation (*Baseline*) constitué d'un modèle de langue général (LM Gen) et d'un modèle de traduction général (TM Gen) sont comparés aux différentes méthodes d'adaptation qui combinent les modèles généraux avec les modèles spécifiques (Spe).

Dans le cas de l'adaptation, chaque phrase est traduite avec le système adapté à sa catégorie. Deux types de systèmes

sont considérés : les premiers entraînés sur des données classifiées automatiquement (colonnes 1, 2, 3 et 5), les seconds entraînés sur des données classifiées manuellement (AFP) (colonne 4). Les phrases classifiées dans la catégorie *Autre* sont traduites avec le système *Baseline* dans lequel les modèles de traduction et de langue sont généraux.

			BLEU				
			Tests spécifiques Class. manuelle			Test initial	
						Class. manuelle	Class. automatique
Approche	TM	LM	POL	WAR	FIN		
Baseline	Gen	Gen	32,08	34,87	31,71	33,47	33,47
(a)	Merge	Gen	30,60	34,68	31,15	32,61	32,41
(b)	Spe+Gen	Gen	31,06	34,26	31,08	32,74	32,41
(c)	Gen	Spe+Gen (log-lin)	31,50	34,62	31,62	32,57	32,85
(b)+(c)	Spe+Gen	Spe+Gen (log-lin)	31,52	33,96	30,43	32,91	32,26
(d)	Gen	Spe+Gen (lin)	**31,94**	**34,84**	**31,83**	**32,98**	**33,14**
(b)+(d)	Spe+Gen	Spe+Gen (lin)	31,37	34,31	31,54	32,85	32,66

TAB. 6 – Traduction automatique en utilisant différentes méthodes d'adaptation, et évaluation sur les tests spécifiques POL, WAR et FIN ainsi que sur le corpus de test initial : chaque phrase est traduite avec le système de traduction spécifique optimisé sur le corpus de développement spécifique et correspondant à la catégorie qui lui est attribuée. Comparaison entre les systèmes construits à partir de modèles spécifiques construits par classification manuelle et les modèles spécifiques construits par classification automatique.

On observe que l'approche (d) - système construit par un modèle de langue interpolé linéairement (lin) et un modèle de traduction général - donne les meilleures performances pour les trois tests spécifiques. On note que le meilleur score pour la classification manuelle du test initial est obtenu également par la même approche.

Une petite amélioration de 0,18 points BLEU pour l'approche (c) et 0,16 points pour l'approche (d) sont observées pour la classification automatique. Bien que les modèles soient plus petits, la performance de traduction n'est pas dégradée. De même avec la classification automatique, les meilleures performances sont données par l'approche (d).

Finalement aucun des modèles adaptés ne permet d'obtenir des performances se traduisant par un meilleur BLEU que le système sans aucune adaptation (Baseline). Ces résultats doivent être relativisés par une étude des sorties de traduction. Les figures 4 et 5 montrent deux exemples de sorties extraites de traductions du corpus de test initial : la phrase en arabe, la référence, la traduction sans adaptation et la traduction avec adaptation en utilisant une catégorisation automatique.

Arabe	[...] ينبغي حل مجلس الامه واحالة النواب الفاسدين علي القضاء
Référence :	le **parlement doit être dissous** et les députés corrompus traduits en justice [...]
Trad. sans adaptation :	il faut une **solution du Conseil de la Nation** et pour traduire en justice les corrompus [...]
Trad. avec adaptation :	il faut la **dissolution du Parlement** et de traduire les députés corrompus [...]

FIG. 4 – Phrase appartenant à la catégorie POL avec sa traduction de référence et les traductions automatiques produites par un modèle sans adaptation et un modèle avec adaptation à la catégorie.

La figure 4 correspond à une phrase appartenant à la catégorie POL. En arabe, le mot حل est ambigu et a pour signification *solution* ou *dissolution* selon le contexte. L'approche avec adaptation permet de résoudre correctement cette ambiguïté ce qui n'est pas le cas pour la traduction sans adaptation. Dans le même temps, le modèle avec adaptation propose une traduction de مجلس الامة plus proche de la référence (*Parlement*) que le modèle sans classification (*Conseil de la Nation*). Ce dernier groupe de mots existe dans le modèle de langue général mais n'existe pas dans le modèle de langue spécifique

à la catégorie POL. Il est à noter également que les mots مجلس et الامة peuvent être traduits indépendamment par *conseil* et *nation*. L'utilisation d'un modèle de langue adapté aide dans ce cas à améliorer la traduction de ce groupe de mots.

L'exemple de la figure 5 permet d'observer également une amélioration d'une sortie de traduction par adaptation à la catégorie. La traduction proposée par le système avec adaptation à la catégorie FIN est plus proche de la référence que celle donnée par le système sans adaptation.

Arabe	وقد انسحبت منها بيونغ يانغ في نيسان / ابريل 2008 [...]
Référence	Pyongyang s' en était officiellement retiré en avril 2008 [...]
Trad. sans classification :	et officiellement , dont Pyongyang a claqué la porte en avril 2008 [...]
Trad. avec class. automatique :	Pyongyang a s' en était retiré officiellement en avril 2008 [...]

FIG. 5 – Amélioration de la traduction (phrase de la catégorie FIN)

On observe donc que, même si cela ne se traduit pas par une amélioration en BLEU, l'adaptation permet dans certains cas d'améliorer les sorties de traduction.

6 Conclusion

Dans cet article, nous avons analysé les résultats d'un ensemble d'expériences d'adaptation thématique pour la traduction automatique. La particularité de notre approche est que nous utilisons un corpus pré-classifié (par les journalistes de l'AFP selon une classification du standard IPTC). Cette classification est utilisée pour construire des modèles spécifiques et comparer plusieurs approches d'adaptation. Deux scénarios de traduction automatique sont proposés : une adaptation reposant sur la *classification manuelle* des phrases et l'autre sur une *classification automatique* en catégorie. Dans les deux cas la traduction est effectuée après la détection des catégories en choisissant le modèle adapté à la catégorie.

Bien que certains exemples montrent que l'adaptation des modèles de traduction à la catégorie permet dans certains cas de désambiguïser la bonne traduction, ceci ne se traduit pas par une amélioration en terme de BLEU.

Deux raisons principales expliquent ces performances décevantes : (i) la trop petite taille des corpus spécialisés, qui conduisent à des modèles probablement plus précis, mais également plus lacunaires ; (ii) le fait que les catégories IPTC (celles utilisées pour cette étude) sont parfois très proches. En particulier, l'effet du manque de données pour estimer les modèles de traduction a un impact important sur les performances finales.

Parmi les 17 catégories IPTC, très peu sont représentées en quantité suffisante dans les données que nous avons traitées. Le choix des catégories (POL, WAR, FIN) a été effectué en se basant sur la quantité de données. Mais ces catégories contiennent beaucoup de phrases communes, ce qui rend leurs frontières floues.

Lors de l'étiquetage manuel du test initial nous avons constaté qu'il est parfois difficile de contraindre l'annotation d'une phrase à une seule catégorie. La phrase ci-dessous (en arabe, avec sa traduction en français) par exemple est extraite d'une dépêche très récente affectée par les journalistes aux catégories ACE (arts, culture et divertissement), POL (politique) et SPO (sport).

عبرت وسائل الاعلام الدولية عن اعجابها بحفل افتتاح دورة الالعاب الاولمبية الشتوية في مدينة سوتشي الروسية، لكن الكثير منها اشار الي الرسالة السياسية التي تنطوي عليها واهميتها في نظر الرئيس فلاديمير بوتين.

Les médias internationaux ont exprimé leur admiration pour la cérémonie d'ouverture des Jeux olympiques d'hiver à la

ville russe de Sotchi, mais beaucoup d'entre eux ont souligné le message politique en cause et son importance aux yeux du président Vladimir Poutine.

S'agissant des jeux olympiques, la phrase ci-dessus doit effectivement être affectée à la catégorie sport. Mais elle traite également du divertissement (cérémonie d'ouverture) et surtout bien sûr de politique. Plusieurs catégories doivent donc être affectées à cette phrase.

Parmi les perspectives pour la suite de nos travaux, il sera en particulier intéressant d'explorer des approches autorisant la multi-catégorisation. Des modèles combinés peuvent être crées et utilisés pour traduire des phrases multi-catégoriques.

Références

BANERJEE P. (2012). *Domain Adaptation for Statistical Machine Translation of Corporate and User-Generated Content*. PhD thesis, Dublin City University.

DE MORI R. & FEDERICO M. (1999). *Language Model Adaptation*, In K. PONTING, Ed., *Computational models of speech pattern processing volume 169, NATO ASI*, p. 280–303. Springer Verlag : Prague, Czech Republic.

EIDELMAN V., BOYD-GRABER J. & RESNIK P. (2012). Topic models for dynamic translation model adaptation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics : Short Papers - Volume 2*, ACL '12, p. 115–119, Stroudsburg, PA, USA : Association for Computational Linguistics.

FOSTER G., GOUTTE C. & KUHN R. (2010). Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, p. 451–459, Stroudsburg, PA, USA : Association for Computational Linguistics.

FOSTER G. & KUHN R. (2007). Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, p. 128–135, Stroudsburg, PA, USA : Association for Computational Linguistics.

GAHBICHE-BRAHAM S., BONNEAU-MAYNARD H., LAVERGNE T. & YVON F. (2012). Joint Segmentation and POS Tagging for Arabic Using a CRF-based Classifier. In *Proc. of LREC'12*, p. 2107–2113, Istanbul, Turkey.

GAHBICHE-BRAHAM S., BONNEAU-MAYNARD H. & YVON F. (2011). Two ways to use a noisy parallel news corpus for improving statistical machine translation. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora : Comparable Corpora and the Web*, p. 44–51, Portland, Oregon : Association for Computational Linguistics.

GAO Q. & VOGEL S. (2008). Parallel implementations of a word alignment tool. In *In* Software Engineering, Testing, and Quality Assurance for Natural Language Processing, p. 49–57.

HADDOW B. & KOEHN P. (2012). Analysing the effect of out-of-domain data on SMT systems. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, Montreal, Canada : Association for Computational Linguistics.

KOEHN P., HOANG H., BIRCH A., CALLISON-BURCH C., FEDERICO M., BERTOLDI N., COWAN B., SHEN W., MORAN C., ZENS R., DYER C., BOJAR O., CONSTANTIN A. & HERBST E. (2007). Moses : open source toolkit for statistical machine translation. In *In* Proceedings of the 45th Annual Meeting of the ACL, p. 177–180.

KOEHN P. & SCHROEDER J. (2007). Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, p. 224–227, Stroudsburg, PA, USA : Association for Computational Linguistics.

LANGLAIS P. (2002). Improving a general-purpose statistical translation engine by terminological lexicons. In *COLING-02 on COMPUTERM 2002 : second international workshop on computational terminology - Volume 14*, COMPUTERM '02, p. 1–7, Stroudsburg, PA, USA : Association for Computational Linguistics.

NAKOV P. (2008). Improving English-Spanish statistical machine translation : experiments in domain adaptation, sentence paraphrasing, tokenization, and recasing. In *Proceedings of the Third Workshop on Statistical Machine Translation*, StatMT '08, p. 147–150, Stroudsburg, PA, USA : Association for Computational Linguistics.

NIEHUES J. & WAIBEL A. (2010). Domain adaptation in statistical machine translation using factored translation models. *Proceedings of EAMT*.

OCH F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, p. 160–167, Stroudsburg, PA, USA : Association for Computational Linguistics.

RIGOUSTE L., CAPPÉ O. & YVON F. (2007). Inference and evaluation of the multinomial mixture model for text clustering. *Information Processing and Management*, **43**(5), 1260–1280.

SENNRICH R. (2012a). Mixture-modeling with unsupervised clusters for domain adaptation in statistical machine translation. In *Proceedings of the 16th EAMT Conference*, p. 185–192, Trento, Italy.

SENNRICH R. (2012b). Perplexity minimization for translation model domain adaptation in statistical machine translation. In W. DAELEMANS, M. LAPATA & L. MÀRQUEZ, Eds., *EACL 2012, 13th Conference of the European Chapter of the ACL, Avignon, France, April 23-27, 2012*, p. 539–549 : Association for Computational Linguistics.

SENNRICH R., SCHWENK H. & ARANSA W. (2013). A multi-domain translation model framework for statistical machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 832–840, Sofia, Bulgaria : Association for Computational Linguistics.

SHAH K., BARRAULT L. & SCHWENK H. (2010). Translation model adaptation by resampling. In *WMT, Association of Computational Linugistics (ACL)*, Uppsala (Sweden).

YAMAMOTO H. & SUMITA E. (2008). Bilingual cluster based models for statistical machine translation. *IEICE Transactions*, **91-D**(3), 588–597.

ZHAO B., ECK M. & VOGEL S. (2004). Language model adaptation for statistical machine translation with structured query models. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA : Association for Computational Linguistics.

Étude quantitative des disfluences dans le discours de schizophrènes : automatiser pour limiter les biais

Maxime Amblard, Karën Fort
Université de Lorraine, LORIA, INRIA, CNRS
UMR 7503, Vandœuvre-lès-Nancy 54500 France
{maxime.amblard, karen.fort}@loria.fr

Résumé. Nous présentons dans cet article les résultats d'expériences que nous avons menées concernant les disfluences dans le discours de patients schizophrènes (en remédiation). Ces expériences ont eu lieu dans le cadre d'une étude plus large recouvrant d'autres niveaux d'analyse linguistique, qui devraient aider à l'identification d'indices linguistiques conduisant au diagnostic de schizophrénie. Cette étude fait la part belle aux outils de traitement automatique des langues qui permettent le traitement rapide de grandes masses de données textuelles (ici, plus de 375 000 mots). La première phase de l'étude, que nous présentons ici, a confirmé la corrélation entre l'état schizophrène et le nombre de disfluences présentes dans le discours.

Abstract. We present in this article the results of experiments we led concerning disfluencies in the discourse of schizophrenic patients (in remediation). These experiments are part of a larger study dealing with other levels of linguistic analysis, that could eventually help identifying clues leading to the diagnostic of the disease. This study largely relies on natural language processing tools, which allow for the rapid processing of massive textual data (here, more than 375,000 words). The first phase of the study, which we present here, confirmed the correlation between schizophrenia and the number of disfluences appearing in the discourse.

Mots-clés : discours pathologique, schizophrénie, disfluences.

Keywords: pathological discourse, schizophrenia, disfluencies.

1 Introduction

1.1 Contexte et motivations de l'étude

Cette étude participe d'un projet plus large portant sur les pratiques langagières chez les schizophrènes en situation d'entretiens semi-dirigés par un psychologue. Cette étude s'inscrit dans la continuité des premiers travaux de Chaika (1974) et Fromkin (1975), qui cherchaient à mettre en avant les particularités langagières chez les schizophrènes.

Plusieurs aspects sont ainsi étudiés, notamment les capacités neuro-cognitives par une série de tests, le comportement oculomoteur du patient par une série d'enregistrements par oculomètre (*eye-tracker*), l'activité encéphale par des enregistrements par électro-encéphalographe (EEG) et la pratique langagière par l'étude linguistique des entretiens. Dans cette partie du projet, nous nous concentrons sur le dernier aspect et laissons donc de côté les autres mesures. Cependant, il conviendra dans une phase ultérieure de revenir sur l'ensemble des analyses pour identifier des corrélations spécifiques.

Concernant la partie linguistique du projet, nous nous basons sur un résultat de psycho-linguistique mettant en avant des usages pathologiques de la langue chez les schizophrènes, au travers de la notion de discontinuités pragmatiques décisives (Musiol & Trognon, 1996; Verhaegen, 2007). Rebuschi *et al.* (2013) et Musiol *et al.* (2013) ont montré que dans la succession des focus thématiques de la conversation, les schizophrènes rejouent une ambiguïté linguistique précédemment introduite, rendant l'interprétation pragmatique et rhétorique impossible. Comme eux, nous souhaitons produire des analyses formelles des extraits discontinus, afin d'en donner une interprétation dans un modèle formel du type SDRT (Asher & Lascarides, 2003) (*Segmented Discourse Representation Theory*), extension à la rhétorique et à la pragmatique de la DRT (Kamp & Reyle, 1993) (*Discourse Representation Theory*). Par ailleurs, il apparaît nécessaire de discuter les

règles du cadre formel car les extraits de dialogue nécessitent l'usage de règles non conventionnelles pour rattraper la construction de telles structures.

Le projet général cherche à ré-interroger ces résultats sur un corpus plus large et sur un faisceaux d'indices diversifiés. D'où, en particulier, l'utilisation des oculomètres et des EEG dans les protocoles. Nous cherchons à interroger également d'autres niveaux linguistiques, en proposant une annotation multi-niveaux de la ressource.

Pour ce faire, la transcription, qui est la clé de voûte de l'ensemble du projet, doit être de qualité. Les outils de transcription automatique que nous avons pu tester ont donné des résultats insuffisants. Elle a donc été réalisé manuellement. Nous avons défini un guide d'annotation précis, dans la tradition de Blanche-Benveniste & Jeanjean (1987). Cependant, l'une des difficultés du projet réside dans la nature des sujets, qui implique une gestion stricte de l'anonymat (voir section 2.3), dont la conséquence est que nous devons minimiser le nombre de personnes ayant accès aux données non transformées. Nous ne disposons donc pour l'instant que d'une seule version des transcriptions, ce qui ne nous permet pas de les évaluer correctement en calculant un accord inter-annotateur.

À partir de ces transcriptions, plusieurs autres annotations vont être proposées, dont une partie va être produite par des outils de traitement automatique des langues (TAL), et une autre par des humains (autre que les transcripteurs). Nous présentons ici le premier niveau d'annotation du corpus, l'annotation en disfluences, réalisée grâce à l'outil `Distagger` (Constant & Dister, 2010). Outre son utilité intrinsèque, l'annotation en disfluences permet de normaliser les corpus avant d'y appliquer des analyseurs syntaxiques ou sémantiques.

1.2 Travaux précédents

Les travaux précédents menés sur le discours des schizophrènes ont donné peu de résultats concernant les disfluences et ces résultats ne sont souvent qu'un élément accessoire de l'expérience décrite. Ainsi, Feldstein (1962) a travaillé sur l'impact du type de contenu d'éléments à commenter (affectif ou non) et a, ce faisant, montré que les perturbations du discours (*speech disturbances*) étaient plus élevées chez les schizophrènes [1]. Ces résultats sont confirmés par la méta étude de Maher (Maher, 1972). Plus récemment, Kremen *et al.* (2003) ont montré, dans le cadre d'une étude concernant la comparaison entre fluence phonémique et fluence sémantique chez les schizophrènes, que ceux-ci ont une fluence verbale (quel que soit son type) légèrement dégradée par rapport aux témoins et aux patients bipolaires [2].

Si les résultats semblent concorder, il n'existe à notre connaissance aucune étude publiée à ce sujet concernant des patients francophones. Par ailleurs, la manière dont les données précédentes ont été annotées ou notées n'est jamais précisée, mais on peut aisément supposer qu'elles l'ont été par des humains, ce qui, en l'absence d'accord inter-annotateurs, est évidemment source de biais [3]. L'étude que nous proposons se singularise donc par l'utilisation d'outils de TAL.

Le présent article est organisé de la manière suivante : nous détaillons dans la section 2 la constitution du corpus et ses implications dans l'étude, tant sur la couverture qu'il propose que sur la délicatesse avec laquelle il est nécessaire de manipuler les données ; puis nous présentons l'outil utilisé pour produire les annotations en disfluences et le protocole d'expérimentation dans la section 3 ; à partir de ces expérimentations, nous analysons dans la section 4 les résultats obtenus, leur significativité et les biais potentiels de l'étude ; enfin, nous présentons les travaux à venir dans la conclusion.

2 Difficultés de constitution du corpus

2.1 Présentation du corpus

Le corpus utilisé pour cette étude est constitué de transcriptions d'entretiens. L'étude fait intervenir 79 sujets, 48 schizophrènes et 31 témoins. Les entretiens ont été réalisés par des psychologues, en milieu hospitalier. Deux recueils de données ont pu être réalisés, dans des unités médicales spécialisées : le premier à Ville1 [4], par deux psychologues, et le second à Ville2, par une seule psychologue.

Le sous-corpus Ville1 a été constitué au second semestre 2013. Il est composé de 18 patients diagnostiqués schizophrènes

1. Cette étude a impliqué 30 schizophrènes et 30 témoins.
2. Cette étude a impliqué 83 schizophrènes, 15 patients bipolaires et 83 témoins.
3. Ce même biais affecte nos transcriptions, mais nos annotateurs ne savaient pas à quoi celles-ci allaient servir, ce qui limite l'impact du biais.
4. Nous avons anonymisé les noms de villes par respect pour la confiance des patients schizophrènes.

en remédiation et sous traitement, ainsi que de 23 témoins. Le sous-corpus Ville2 a été constitué au printemps 2002. Il est composé de 30 patients diagnostiqués schizophrènes en remédiation et sous traitement, à l'exception de sept d'entre eux (qui n'étaient pas sous traitement), et de huit témoins. Le tableau 1 présente la ventilation des sujets en fonction de leur type (schizophrène ou témoin) et de leur sexe.

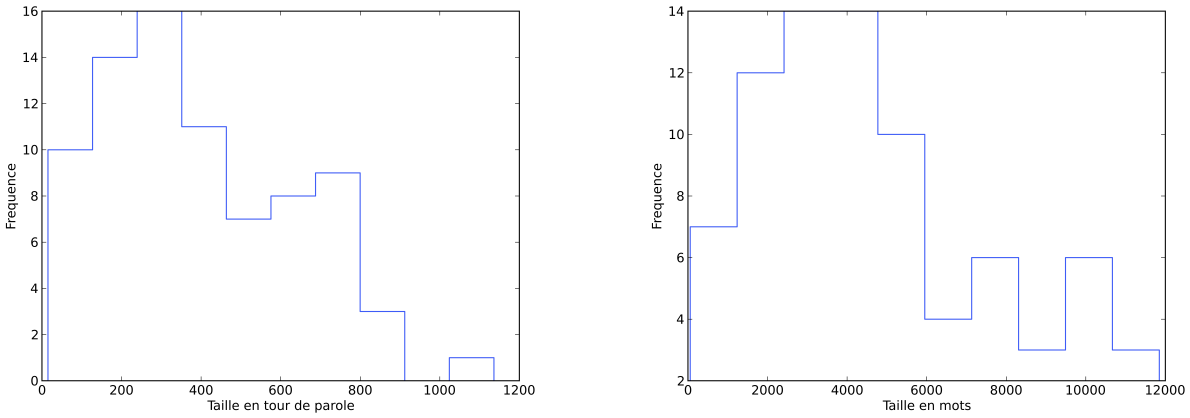
	corpus Ville1			corpus Ville2			total
	hommes	femmes	total	hommes	femmes	total	
schizophrènes	15	3	18	20	10	30	48
témoins	15	8	23	4	4	8	31
total	30	11	41	24	14	38	79

TABLE 1 – Répartition des sujets dans le corpus.

L'interaction mise en place pour cette étude est un entretien semi-directif conduit par un psychologue. Dans ce type d'entretien, le psychologue n'est pas personnellement engagé dans l'interaction. Il doit maintenir un échange dans lequel le patient revient sur son environnement et ses relations au sein de l'hôpital et avec l'extérieur. Il est clairement expliqué, tant à l'équipe médicale qu'au patient, que le contenu de l'entretien ne peut être utilisé comme base médicale.

Le protocole expérimental a consisté à identifier des patients intéressés, puis à leur faire passer des tests de mesure de capacité cognitive. Dans le sous-corpus de Ville2 aucune mesure supplémentaire n'a été faite, dans le sous-corpus de Ville1, les entretiens ont eu lieu en présence d'un double système d'oculomètre [5].

Le protocole a été défini de manière à être le moins invasif possible. Pour le sous-corpus Ville1, trois tests psychocognitifs mesurant les capacités de mémoire à court terme, d'attention, et la mémoire de travail ont été passés par les sujets : (i) le `Wechsler Adult Intelligence Scale-III` (mesure du quotient intellectuel, ou QI), (ii) le `California Verbal Learning Test` (capacité cognitive et de stratégie), et (iii) le `Trail Making Test` (dépréciation de la flexibilité cognitive et de l'inhibition). Nous n'utiliserons ici que les résultats du test de QI.



(a) (b)

FIGURE 1 – Distribution des entretiens par la taille : (a) en nombre de tours de parole ; (b) en nombre de mots.

Une fois les entretiens enregistrés, ils ont été transcrits manuellement par deux annotateurs (le ou la psychologue qui a mené tout ou partie des entretiens, et une autre personne) qui n'ont pas annoté en parallèle et qui ne savaient pas que nous allions compter les disfluences. En moyenne, les entretiens du sous-corpus de Ville1 sont constitués de 552,73 tours de parole, alors que les entretiens du sous-corpus Ville2 en contiennent 234,5. L'ensemble du corpus comprend 31 575 tours de parole, soit environ 375 000 mots. Le tableau 2 présente le découpage en tours de parole et en mots du corpus, et la figure 1 illustre la distribution des corpus en fonction de leur taille en nombre de tours de parole et de mots. Le caractère spécifique de l'entretien semi-directif transparaît ici clairement : le psychologue produit quasiment le même nombre de

5. Ce double système permet de capter les points de fixation du regard du sujet sur 'ce qu'il voit', mais également de capter ceux du psychologue. Ainsi, il est possible d'interpréter si un mouvement est déclenché par une interaction visuelle (regard de l'interlocuteur) ou non.

	corpus Ville1				corpus Ville2			
	nb tours de parole		nb de mots		nb tours de parole		nb de mots	
S	3 863	} 11 145	46 859	} 119 762	4 062	} 4 433	66 725	} 79 081
T	7 282		72 903		371		12 356	
$P+S$	3 819	} 11 517	30 293	} 138 571	4 098	} 4 480	33 686	} 37 842
$P+T$	7 698		108 278		382		4 156	
total	22 662		258 333		8 913		116 923	

TABLE 2 – Décomposition du corpus en sous-corpus, en nombre de tours de parole et nombre de mots, en fonction du type d'interlocuteur. S (schizophrènes), T (témoins), P + S (psychologue avec un schizophrène), P + T (psychologue avec un témoin).

tours de parole que le sujet, pour un volume de mots très inférieur. Par exemple, dans le sous-corpus Ville1, le ratio entre le nombre de tours de parole des schizophrènes et des psychologues devant un schizophrène est de 1,003 (seulement 44 tours de parole), alors qu'il est de 1,54 en nombre de mots. Seuls les témoins du sous-corpus de Ville1 ne présentent pas cette caractéristique, mais une analyse plus fine des entretiens montre que pour six entretiens, les témoins sont restés réticents à prendre la parole.

2.2 Difficultés d'accès aux patients

Le nombre de 79 sujets peut sembler limité pour une étude de ce type, mais la constitution d'une telle ressource implique de surmonter de nombreuses difficultés, en particulier pour accéder aux patients. De ce fait, disposer d'une cinquantaine de transcriptions d'entretiens avec des schizophrènes représente un corpus significatif.

Pour s'entretenir avec une personne prise en charge par le milieu hospitalier, il est en effet nécessaire d'obtenir une autorisation du CPP (Comité de Protection de la Personne) de la région de l'établissement. Les demandes déposées doivent contenir explicitement et exactement le contenu du protocole de test. L'instruction du dossier requiert plusieurs mois et elle demande la contraction d'une assurance pour prendre en charge les possibles dommages. De fait, ce dernier point augmente considérablement les budgets nécessaires pour ce type d'expérience. Une fois les accords obtenus, il n'est alors plus possible de modifier les protocoles.

Mais ce qui rend la constitution d'une telle ressource complexe est principalement la difficulté de faire participer les patients. Plusieurs problèmes se posent. Il faut d'abord identifier, au sein d'un service, les patients répondant aux critères de l'étude en capacité d'interagir avec une personne tierce au service. Puis il faut, au sein de cette population, trouver les patients qui acceptent de participer à l'étude. Une première réticence vient du fait qu'il n'y a pas de conséquence positive, en terme médical, pour le patient à participer à l'étude. Il faut ajouter à cela des inquiétudes compréhensibles des patients schizophrènes concernant la possible publication de leur histoire, bien qu'une anonymisation totale soit garantie par le protocole. Bien entendu, la sous-catégorie des schizophrènes paranoïdes est encore plus difficile à rencontrer.

Par ailleurs, le protocole requérant de passer des tests psycho-cognitifs et un entretien, le temps nécessaire est relativement élevé, de l'ordre de deux heures. Ce n'est pas tant la disponibilité des patients qui est alors en jeu, que leur aptitude à rester concentrés. Lorsque le patient présente soudainement des difficultés, il faut convenir d'un second rendez-vous pour finaliser le protocole. La multiplication des rendez-vous génère également des défections. À titre d'exemple, lors de la phase de collecte des entretiens du sous-corpus Ville1, 45 % (18) des patients contactés ont refusé de participer, 10 % ont accepté un premier rendez-vous mais ne sont pas présentés au second, et 45 % (18 sujets) ont participé à toute l'étude.

2.3 Anonymisation

La tâche d'anonymisation recouvre deux phases. La première, tout à fait classique, consiste à identifier les entités nommées et à les substituer par des marqueurs sémantiquement vides. Un outil automatique performant a été identifié pour ce faire, mais n'a pu être opérationnel à temps pour cette étude. Nous avons pour cela programmé une série de scripts en Python qui recherchent, grâce à des expressions régulières, les mots commençant par une majuscule qui ne sont pas

en début de phrase. Une intervention humaine a été ensuite nécessaire pour classer ces mots en 10 catégories : *prenomF*, *prenomM*, *nom*, *pays*, *département*, *ville*, *capitale*, *institution*, *montagne* et *non_pris_en_compte*. Les éléments de cette dernière catégorie ont été laissés tels quels dans le corpus, les autres ont été substitués par le nom de la catégorie suivie d'un identifiant unique. Ainsi, les références à Paris sont toutes identifiées par *capitale1*. Une fois ces substitutions réalisées, nous avons extrait l'ensemble des débuts de phrases et procédé à une vérification manuelle pour affiner les listes des catégories précédentes. Nous pouvons ainsi assurer une anonymisation fiable du corpus.

L'anonymisation du corpus ne s'arrête cependant pas là. En effet, les sujets relatant des événements s'inscrivant dans une temporalité et une géographie particulière, un certain nombre d'indices sont disséminés dans les entretiens. Il est donc relativement aisé d'identifier les personnes et il est difficile de trouver une solution à ce problème tout en conservant l'intégrité des entretiens. Cette particularité a des conséquences importantes sur notre projet.

Pour les traitements qui ne nécessitent qu'un faible contexte, en général celui de la phrase ou du tour de parole, nous avons créé une version de la ressource constituée de tous les tours de paroles randomisés. Les 31 575 tours de paroles sont donc mélangés et il devient impossible de reconstituer les historiques de chacun. Pour reconstruire les entretiens originaux, nous conservons une trace de la randomisation sous forme de table. Il est donc tout à fait possible de fournir la ressource pour des analyses du type morpho-syntaxe ou analyse syntaxique en dépendances, sans compromettre les données initiales.

Mais l'un des objectif du projet global reste l'analyse sémantico-pragmatique et, pour ces aspects, il est impossible de dissocier une prise de parole de son contexte sans perdre l'essence même de l'entretien. Seuls les membres engagés dans le projet et soumis à un devoir de confidentialité peuvent donc travailler sur cette partie. Un problème similaire se pose pour la partie transcription, puisque, bien que les bandes puissent être bippées, elles ne peuvent pas être randomisées en tours de parole. Cette contrainte explique que le nombre d'intervenants sur la transcription reste limité.

3 Protocole expérimental

3.1 Traitements automatiques

Étant donnée la taille importante des corpus et notre volonté de limiter les interventions humaines, nous avons utilisé l'outil `Distagger` (Constant & Dister, 2010) pour identifier automatiquement les disfluences dans les textes transcrits.

Pour cet outil, les disfluences regroupent plusieurs types de réalisations orales qui brisent la continuité syntaxique. Il est donc possible de produire une version reconstruite de la ressource pour obtenir des tours de parole plus cohérents du point de vue syntaxique, donc améliorer les résultats des annotations pour d'autres couches (en particulier l'analyse morphosyntaxique ou en dépendances).

L'outil permet d'identifier des réalisations de natures différentes, pour lesquelles quatre restent prédominantes dans les corpus oraux : les *euh*, les répétitions, les autocorrections immédiates et les amorces de morphèmes. Nous revenons sur chacune d'elles en présentant un exemple extrait du corpus.

1. Les différentes réalisations de *euh* sont définies dans un fichier passé en argument de `Distagger`.

   (1) moi ça m'est presque plus euh difficile et euh anti-naturel de parler

2. Les répétitions sont entendues comme la reprise explicite et identique d'un même mot ou d'un même groupe de mots dans le contexte immédiat d'apparition. La répétition peut malgré tout contenir ou être précédée d'un mot creux comme *oui*, *non*, ou un *euh* :

   (2) j' arrive à être à être concentrée quand il faut faire quelque chose

3. L'autocorrection immédiate est une variante de la répétition dans laquelle un trait morphologique peut varier (ce qui apparaît régulièrement avec les déterminants) :

   (3) enfin je sais pas trop le les termes

4. L'amorce est une interruption de morphème en cours d'énonciation. La fin du mot est marquée par un -.

   (4) pis progressivement vous av- pouvez travailler sur votre concentration

Les auteurs ont évalué leur outil sur un corpus oral marqué en disfluences et validé manuellement. Le corpus de référence comprend au total 1 297 tours de parole, 22 476 mots, 5 817 méta-étiquettes et 1 280 disfluences. Ils obtiennent des f-scores significatifs, de 95,5 % (précision de 95,3 %, rappel 95,8 %) [6].

L'outil prend en entrée des données au format `Valibel` [7] (sans structure prédéfinie) ou au format `transcriber` (Barras *et al.*, 1998) (structuré et semi-annoté). Il fournit deux types de sortie, l'un correspondant au format `Valibel`, l'autre au format `transcriber`. `Distagger` est implémenté en Java, et peut être appelé en ligne de commande, ce qui nous a permis de l'intégrer facilement à notre chaîne de traitement. Par ailleurs, l'outil ajoute plusieurs annotations particulières structurant son résultat, qui ne sont pas informatives pour les disfluences.

Les annotations de `Distagger` sur le corpus font apparaître sept étiquettes : $\{IGN+EUH\}$, $\{IGN+REP\}$, $\{IGN+CORR\}$, $\{IGN+FRAG\}$, $\{IGN+short\_pause\}$, $\{IGN+slot\}$ et $\{IGN+speaker\}$. Les deux dernières sont des étiquettes spécifiques permettant de repérer les tours de parole et les interlocuteurs à qui sont associés ces tours de parole. Leur nombre n'apporte pas d'information caractéristique ici et elles seront écartées dans la suite. Par ailleurs, les premiers traitements ont fait apparaître les étiquettes $\{IGN+short\_pause\}$ et $\{IGN+FRAG\}$ dans des volumes très faibles (respectivement 5 et 1 étiquettes). Les $short\_pause$ correspondent à des reliquats de scories de la transcription qui ont été mal interprétés par l'outil.

Nous avons par ailleurs mis en place une série de programmes en Python pour pré-traiter les corpus, appliquer `Distagger` et post-traiter les résultats fournis.

3.2 Normalisation des corpus

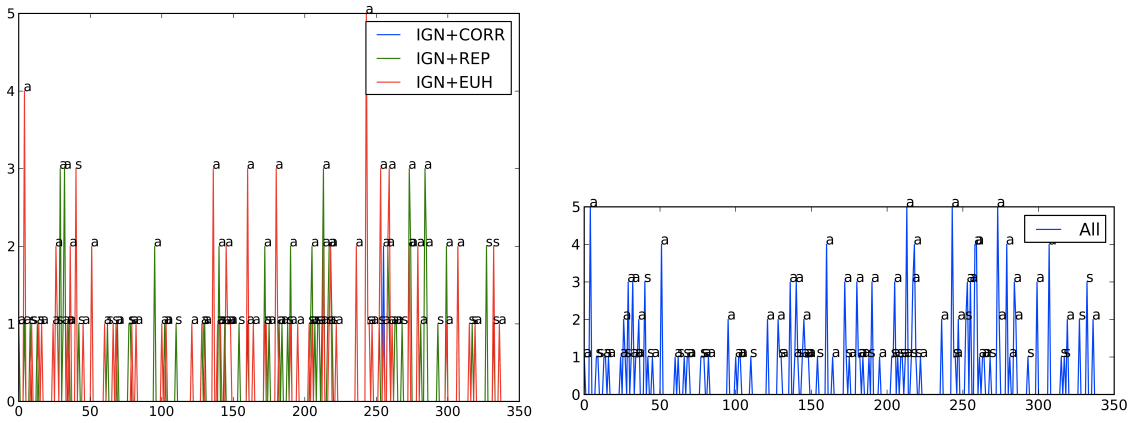
Le sous-corpus Ville2 initial n'ayant pas été prévu pour être traité par des outils de TAL, une première étape a donc consisté à extraire le contenu des documents en format `MS Word` et à normaliser le corpus à l'aide d'une trentaine d'expressions régulières. Il a fallu réinterpréter les marques spécifiques à la transcriptions originelle vers des marques explicites pour `Distagger` (↑ pour une intonation montante, ↓ pour une intonation descendante, etc.). Cinq traitements sont nécessaires pour ajouter aux fichiers les informations permettant à `Distagger` de fonctionner (utilisation de $spk1$ et $spk2$ pour les interlocuteurs [8], chemin explicite des fichiers, etc.). Ainsi, pour chaque fichier du sous-corpus de départ, nous obtenons sa version annotée par `Distagger`.

Puis, l'ensemble des résultats obtenus est fusionné pour produire une représentation générale pour ce sous-corpus, en associant à chaque tour de parole le numéro du corpus (ici 1), l'identifiant du sujet (deux chiffres et trois lettres), le numéro du tour de parole dans l'entretien, ainsi qu'une marque explicite de qui est l'interlocuteur (Pa pour le patient, et P pour le psychologue). Enfin, il manque une information discriminante qui est le statut du sujet : schizophrène ou témoin. Pour cela, une base qui distingue entre les deux, et dont la hiérarchie est fixée préalablement, est automatiquement produite à partir de la structure du corpus de départ. Nous construisons alors une représentation abstraite du corpus à partir de la fréquence d'apparition des différentes étiquettes de `Distagger`.

Le sous-corpus Ville1 s'inscrit dans le cadre du projet général. Les annotateurs ont utilisé l'outil `CLAN` pour réaliser la transcription, ce qui nous a permis d'extraire facilement le contenu textuel. Cependant, de nombreuses marques dépendantes du logiciel perdurent et il a été nécessaire de les supprimer. À nouveau, une trentaine d'expressions régulières a été utilisée. Comme pour Ville2, nous produisons une ressource contenant l'ensemble des tours de parole randomisé. C'est sur cette ressource que `Distagger` est utilisé. Puis, en utilisant la table de mémorisation, les entretiens sont reconstruits et envoyés à la même série de traitements que le corpus Ville2. Enfin, pour faciliter les études sur le contenu des entretiens, chacun est mémorisé sans aucune autre marque que l'interlocuteur (*P* ou *S* ou *T*) [9].

Il apparaît dans notre corpus que certaines annotations ne correspondent pas à des disfluences traditionnelles. En effet, le psychologue ayant une interaction particulière dans l'entretien, puisque son rôle est d'abord de maintenir l'échange, il utilise régulièrement des interventions de type *mmh mmh*, ou *oui oui*, ou *non non*. Nous avons donc mis en place un post-traitement qui redresse les résultats en supprimant les étiquettes correspondantes. De plus, la forme pronominale est très fréquemment utilisée, puisque les interlocuteurs se vouvoient et que le psychologue relance la conversation en posant des questions à caractère personnel. Nous avons donc inclus à ces post-traitements les formes *vous vous* qui apparaissent

6. Une évaluation de l'outil sur un échantillon de données (4 entretiens) a mis au jour un taux d'erreur compris entre 5 et 10 %. Une analyse de ces erreurs a montré qu'elles étaient majoritairement dues à des interruptions mal identifiées, problème que nous avons corrigé depuis.
7. Voir : `https://www.uclouvain.be/cps/ucl/doc/valibel/documents/conventions_valibel_2004.PDF`.
8. $spk1$ est attribué au premier locuteur et ne correspond pas nécessairement au psychologue.
9. Il nous semble important de prendre en considération les disfluences du psychologue qui influencent le cours de l'interaction.



(a) (b)

FIGURE 2 – Distribution des étiquettes de disfluence au cours de l'entretien 011HAM du sous-corpus Ville1 (a est un tour de parole du patient et s, du psychologue).

en nombre très supérieur par rapport aux répétitions de *vous*.

Enfin, nous avons automatisé la production des graphiques pour chaque entretien, reprenant le nombre de disfluences par tour de parole, comme présenté dans la figure 2. Pour chaque sous-corpus nous produisons une surface où chaque entretien est ramené à l'échelle en pourcentage (le nombre de tours de parole fluctuant beaucoup) en fonction du nombre de disfluences présent dans la portion de texte (voir figure 3. Enfin, nous produisons plusieurs documents LaTeX reprenant les moyennes des disfluences par tour de parole et par nombre de mots, et nous calculons pour chaque sous-corpus leur indice de significativité. Nous produisons ensuite une représentation graphique de la position des disfluences dans l'entretien. L'ensemble de ces programmes de normalisation représente environ 1 500 lignes de code.

4 Résultats

4.1 Analyse quantitative

La figure 2 présente un exemple des résultats obtenus, pour le patient 011HAM (sous-corpus Ville1). Dans la première figure (a), une couleur de courbe est attribuée à chacune des trois étiquettes principales, l'axe des abscisses correspond aux tours de parole de l'entretien et celui des ordonnées au nombre de disfluences dans ce tour de parole. Pour les points où l'ordonnée est différente de 0, une étiquette est ajoutée, a pour les tours de paroles du patient, et s pour le psychologue. La seconde figure présente les mêmes données, en exhibant la somme du nombre d'étiquettes pour le même tour de parole. Ainsi, la première valeur significative est à 4 dans la figure a et à 5 dans la figure b. Il se trouve que dans ce tour de parole `Distagger` identifie 4 *euh* et 1 *rep*, ce qui explique la différence observée.

Une lecture de l'ensemble des graphiques fait apparaître une régularité avec deux pics de disfluences, le premier en début d'entretien, le second au cours du dernier tiers. Le premier pic peut simplement s'entendre comme un pic de stress en début d'échange. Il est intéressant de constater que le second pic amène la fin de l'entretien. Du côté des témoins, on retrouve une entame d'entretien avec un pic, mais pas nécessairement un second.

La table 3 présente l'ensemble des résultats de `Distagger`. Pour chacun des deux sous-corpus nous calculons, pour les trois étiquettes principales ($IGN{+}CORR$, $IGN{+}REP$ et $IGN{+}EUH$), leur fréquence d'apparition dans les tours de parole, des schizophrènes (S), des témoins (T) ou du/de la psychologue (P). Nous normalisons d'une part par rapport au nombre de tours de parole (pour chaque catégorie d'interlocuteur), et d'autre part par rapport au nombre de mots (à nouveau pour chaque type d'interlocuteur). Nous calculons les mêmes valeurs pour les sujets (S + T), le ou la psychologue lorsqu'il ou elle est en face d'un schizophrènes (P+S) ou devant un témoin (P+T). Les résultats totaux reprennent la

	corpus Ville2						corpus Ville1					
	S	T	S+T	P+S	P+T	P	S	T	S+T	P+S	P+T	P
IGN+CORR												
par tour de parole	0,0087	0,0071	0,008	0,0015	0	0,0012	0,0180	0,0085	0,0127	0,0052	0,0109	0,0084
par / nb mots	0,0004	$9e-05$	0,0003	0,0001	0	0,0001	0,0013	0,0007	0,0010	0,0006	0,0007	0,0006
IGN+REP												
par tour de parole	0,2223	0,2519	0,2285	0,0646	0,0897	0,0699	0,2735	0,138	0,1978	0,1336	0,2608	0,205
par / nb mots	0,0125	0,0078	0,0115	0,0064	0,0079	0,0067	0,0211	0,0134	0,0168	0,0171	0,0177	0,0174
IGN+EUH												
par tour de parole	0,3107	0,2999	0,3084	0,0738	0,0616	0,0712	0,4201	0,3372	0,3736	0,1948	0,4651	0,3464
par / nb mots	0,0190	0,0089	0,0169	0,0077	0,0058	0,0073	0,0369	0,0326	0,0345	0,0244	0,0312	0,0282
Resultats totaux												
par tour de parole	0,5417	0,5589	0,545	0,1400	0,1513	0,1424	0,7117	0,484	0,5842	0,3338	0,7369	0,5599
par / nb mots	0,032	0,0168	0,0288	0,0144	0,0138	0,0142	0,0595	0,0468	0,0524	0,0421	0,0496	0,0463

TABLE 3 – Répartition quantitative des étiquettes de `Distagger` dans les sous-corpus.

somme des valeurs intermédiaires pour chaque catégorie d'interlocuteur. `Distagger` n'annote aucune correction pour le psychologue avec des témoins dans le sous-corpus Ville2, ce qui explique les deux valeurs à zéro.

La lecture des résultats totaux met en avant une variabilité importante des résultats normalisés par rapport au nombre de tours de parole. Les résultats normalisés par rapport au nombre de mots sont plus significatifs. En effet, si les disfluences produites par les témoins et le psychologue (quel que soit son interlocuteur) sont du même ordre : $1,68$ % et $1,42$ % pour le sous-corpus Ville2, et $4,68$ % et $4,63$ % pour le sous-corpus Ville1, les productions des schizophrènes sont bien supérieures : $3,2$ % et $5,95$ %. Il existe ainsi une différence entre le nombre de disfluences identifiées chez les schizophrènes et les non schizophrènes de $1,63$ % dans le sous-corpus Ville2 et de $1,29$ % dans le sous-corpus Ville1.

La variabilité des résultats peut s'expliquer par la différence des transcriptions entre les deux sous-corpus, ainsi que par le nombre de sujets dans chacun. S'il n'est pas raisonnable de proposer le calcul d'un résultat pour l'ensemble du corpus, la constance de la différence de résultats conduit à notre conclusion.

Enfin, pour visualiser la répartition des résultats, nous produisons une surface où nous normalisons le nombre de tours de parole sur une échelle de 100. Pour chaque entretien seuls les tours de parole, soit du schizophrène, soit du témoin, sont utilisés et nous recalculons le nombre de disfluences sur un intervalle de 1 % de l'entretien. Nous obtenons ainsi des pics où, en valeur, le nombre est supérieur aux résultats précédents, mais qui correspondent à une combinaison linéaire de plusieurs tours de parole. La figure 3 présente les surfaces calculées pour les deux sous-corpus. Les deux figures de gauche correspondent au sous-corpus de Ville1, les deux figures de droite à celui de Ville2. Les figures en haut correspondent aux tours de parole des schizophrènes et celles en bas à ceux des témoins.

Dans le cas de Ville2, il apparaît de manière évidente que les témoins produisent beaucoup moins de disfluences que les schizophrènes. La surface en bas à gauche est en effet quasiment plane. L'interprétation des graphiques pour le sous-corpus Ville1 est plus délicate. Pour cela, nous présentons la projection de l'ensemble des entretiens sur l'axe *pourcentage* du graphique. La couleur bleu correspond à une densité importante sur la projection et la couleur rouge à une densité plus faible. En étudiant les valeurs ainsi trouvées pour le sous-corpus, on trouve une distribution régulière chez les témoins, mais toujours plus marginale que chez les schizophrènes.

4.2 Significativité

Afin de valider les résultats que nous avons pu mettre en avant, nous reprenons ici la mesure de significativité utilisée dans (de Mareüil *et al.*, 2013). Celle-ci permet de calculer un indice de distribution en fonction du nombre de mots entre deux catégories d'interlocuteurs. La valeur trouvée doit être supérieure à 1,96 pour être considérée comme significative.

$$s = \frac{(p_1 - p_2)}{\sqrt{p(1-p)(\frac{1}{n_1} + \frac{1}{n_2})}}$$

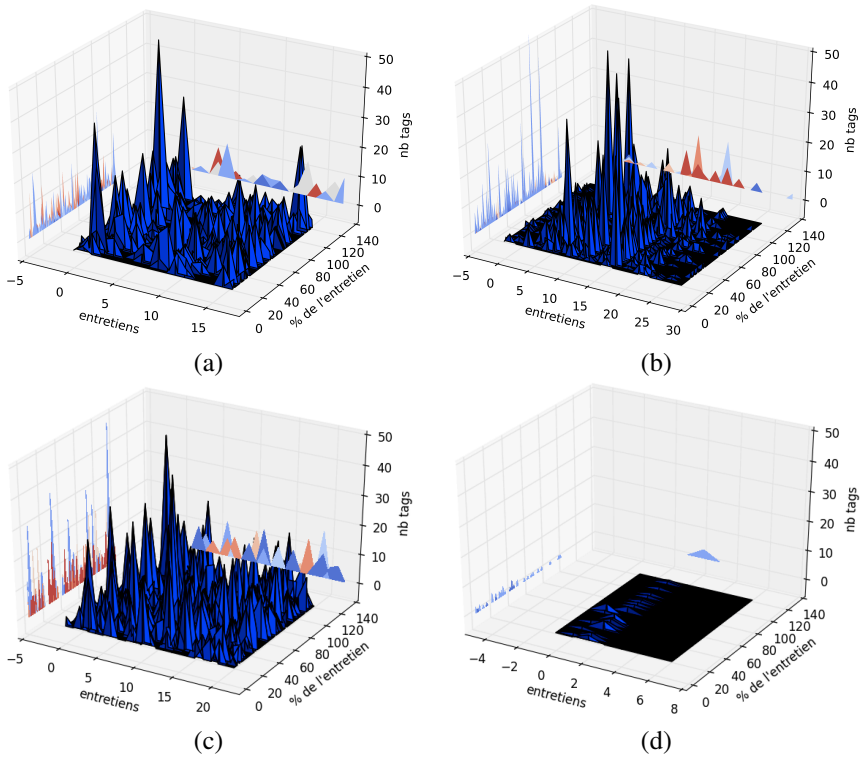


FIGURE 3 – Répartitions des étiquettes de disfluences : dans les entretiens de schizophrènes, sous-corpus Ville1 (a), Ville2 (b) ; dans les entretiens de témoins, sous-corpus Ville1 (c), Ville2 (d).

où :

$$p = (n_1 p_1 + n_2 p_2)/(n_1 + n_2)$$

– n_1 est le nombre de mots prononcés par la première catégorie d'interlocuteur,
– n_2 est le nombre de mots prononcés par la seconde catégorie d'interlocuteur,
– p_1 est la proportion de disfluences produites par la première catégorie d'interlocuteur,
– p_2 est la proportion de disfluences produites par la seconde catégorie d'interlocuteur.

Cette formule nous permet d'interpréter les résultats entre deux catégories d'interlocuteurs. Nous calculons donc l'indice pour les trois appariements que nous pouvons proposer. Les résultats obtenus sont présentés dans le tableau 4.

	corpus Ville1	corpus Ville2
S et Psy	$10,6806923083$	$19,4197596818$
T et Psy	$0,422898291704$	$3,23530253756$
S et T	$10,2827554261$	$16,0376100956$

TABLE 4 – Significativité des différences dans le nombre de disfluences entre interlocuteurs.

Il apparaît que les significativités entre les témoins et les psychologues sont faibles, voire non significatives, ce qui permet de rapprocher le comportement des témoins de celui des psychologues. Par contre, la significativité est importante (toujours supérieure à 10) dans les appariements qui comprennent des schizophrènes, ce qui nous permet de conclure que le nombre de disfluences produites par des schizophrènes est significativement différent de celui des non-schizophrènes de l'expérimentation (psychologue et témoins). Ce résultat est d'autant plus pertinent qu'il est mis en avant par des outils automatiques et ne fait pas intervenir de subjectivité humaine.

	nb étiquettes	/nb mots	/ nb tours de parole
$IGN{+}ENR$	2	$2e-05$	$0,00037$
$IGN{+}REP$	2 208	$0,01857$	$0,40314$
$IGN{+}Meta$	161	$0,00135$	$0,0294$
$IGN{+}EUH$	2 773	$0,02333$	$0,5063$
$IGN{+}CORR$	138	$0,00116$	$0,0252$

TABLE 5 – Analyse des disfluences dans le corpus TCOF-POS par `Distagger`.

4.3 Biais potentiels des expériences

Malgré la significativité des résultats que nous avons obtenus, il nous paraît important de revenir sur plusieurs biais potentiels de l'étude.

Les deux sous-corpus ayant été transcrits par des moyens différents et jamais en parallèle, comme nous l'avons discuté dans la section 2, il est difficile de proposer une évaluation qualitative des transcriptions. Afin de vérifier que nos résultats ne dévient pas de la réalité linguistique, nous avons appliqué `Distagger` sur le corpus de parole spontanée TCOF-POS (Benzitoun *et al.*, 2012). La table 5 reprend la ventilation des étiquettes trouvées par `Distagger`. Sur les trois étiquettes que nous retrouvons et avons analysé, le nombre de disfluences est de $4,3$ %, ce qui est comparable aux résultats précédents et nous conduit à considérer la transcription comme un faible biais.

Un autre biais réside dans la répartition entre témoins et patients à l'intérieur même de chaque sous-corpus. Ainsi, le sous-corpus Ville2 ne contient que 8 témoins, alors que le sous-corpus Ville1 en contient 23. Nous avons décidé de conserver tous les témoins dont nous disposions pour équilibrer le corpus général. Le projet s'attache à rééquilibrer la répartition. Les témoins du sous-corpus Ville1 produisent davantage de disfluences que ceux du sous-corpus Ville2. Une lecture des entretiens montre que la psychologue qui a recueilli les entretiens du sous-corpus Ville1 produit plus de disfluences, ce qui peut inciter les interlocuteurs à l'imiter, donc à produire davantage de disfluences.

Il existe par ailleurs une différence d'âge et de QI entre les participants schizophrènes et témoins (voir tableau 6). En effet, les schizophrènes sont significativement plus âgés que les témoins (près de 29 ans au lieu de 23, avec $p = 0,0058$ [10]) et leur QI est inférieur (à peu près 95 au lieu de 103, avec $p = 0,0203$), pour un nombre d'années d'école ou d'études très semblable (environ 13 pour les témoins et 12,4 pour les schizophrènes).

	QI	années d'études	âge
femmes			
témoins	$105,5$	13	$22,37$
schizophrènes	$98,33$	13	30
hommes			
témoins	$102,73$	$13,26$	$23,66$
schizophrènes	$94,53$	$12,28$	$28,66$
moyenne générale			
témoins	$103,70$	$13,17$	$23,22$
schizophrènes	$95,17$	$12,41$	$28,89$

TABLE 6 – Moyennes des QI, du nombre d'années d'études et des âges sur les participants au corpus Ville1.

Un autre biais important, mais inévitable, de l'étude est que les patients sont en remédiation, donc sous traitement (Chlorpromazine à Ville2 et neuroleptiques non spécifiés à Ville1). Levy (1968) a identifié des effets négatifs (en l'occurrence, une baisse des performances) de la Chlorpromazine sur la syntaxe de quatre patients schizophrènes en calculant le ratio du nombre de propositions subordonnées produites sur la totalité des propositions produites. En outre, cet antipsychotique semble provoquer des bégaiements (Ward, 2008). Cependant, Goldman-Eisler *et al.* (1965) a montré (sur des sujets non schizophrènes) que les effets de cette même molécule sur les temps de pause du locuteur sont très variables selon les individus et qu'un temps de pause supérieur permet au groupe testé de générer des structures verbales complexes,

10. Les significativités ont ici été calculées à l'aide du test de Student.

comme chez les témoins. Pour ajouter à ces incertitudes, Kremen *et al.* (2003) ont montré que des patients bipolaires sous antipsychotiques (dont fait partie la Chlorpromazine) présentent une meilleure fluence sémantique que les témoins. Il est donc aujourd'hui extrêmement difficile d'évaluer l'influence exacte du traitement sur les productions des patients. Cette question est récurrente dans la littérature. Il apparaît néanmoins que les effets secondaires des médicaments sont moins prégnants aujourd'hui, les traitements ayant considérablement évolué depuis les années 60. Par ailleurs, nous disposons d'un sous-groupe de 7 patients schizophrènes sans traitement dans le sous-corpus Ville2. Les différences entre eux et les autres patients schizophrènes n'apparaissent pas significatives.

5 Conclusions et perspectives

Cette étude nous a permis de mettre en lumière un usage pathologique des disfluences chez les patients schizophrènes grâce à des outils et des méthodes issus du TAL. Pour cela, nous avons utilisé l'outil `Distagger` pour procéder à une annotation des disfluences. Il apparaît que les schizophrènes produisent, respectivement dans chaque corpus, 1,63 % et 1,29 % de disfluences de plus (par rapport au nombre de mots) que des sujets non diagnostiqués. Nous avons validé ce résultat par un calcul de significativité qui isole clairement les patients schizophrènes. Ce sont les outils de TAL qui ont permis d'aboutir à cette conclusion, en dehors de toute interprétation humaine.

Nous avons par ailleurs discuté des différents biais possibles de l'étude, tant sur la constitution du corpus de départ que sur la méthodologie utilisée. La suite du projet s'attachera à revenir sur ces derniers pour les corriger lorsque cela est possible, en particulier en calculant une mesure de qualité des transcriptions. Mais l'une des difficultés principales réside dans la nature de l'objet d'étude. D'une part, les patients schizophrènes doivent être suffisamment communiquant pour passer le protocole de tests et donc, généralement, être sous traitement. D'autre part, le caractère personnel de l'entretien pose plusieurs questions d'éthique. Par conséquent, si il est difficile d'accéder aux patients, il est tout aussi délicat de gérer l'accès à la ressource.

Dans la continuité du projet, nous souhaitons annoter la ressource en morpho-syntaxe et en syntaxe en dépendances. L'annotation en morpho-syntaxe nous permettra de réaliser automatiquement une lemmatisation du corpus, notamment pour évaluer la richesse du vocabulaire des sujets. Pour cela, nous allons utiliser l'outil `MElt` (Denis & Sagot, 2009) entraîné pour le français oral sur le corpus TCOF-POS (Benzitoun *et al.*, 2012) [11]. L'analyse en dépendances nous permettra de revenir sur les résultats anciens auxquels nous avons fait référence sur la complexité de la syntaxe utilisée par les patients. Pour cela, plusieurs outils sont disponibles, dont `FRMG` (De La Clergerie *et al.*, 2009), `Leopar` (Perrier & Guillaume, 2013) et `Talismane` (Urieli & Tanguy, 2013), plusieurs tests préliminaires ont d'ailleurs été d'ores et déjà réalisés. L'utilisation de plusieurs outils nous permettra de valider les analyses proposées.

Comme il a été fait mention dans la première partie de cet article, l'objectif est également de proposer une annotation en sémantique-pragmatique. Pour cela, nous conduirons deux campagnes d'annotation manuelle, l'une pour identifier les discontinuités décisives, l'autre, sur les extraits identifiés, pour annoter en SDRT. L'ensemble de ces indices seront corrélés avec les autres mesures dont nous disposons, dont les résultats aux tests psycho-cognitifs, les mesures oculométriques et les EEG.

La contribution apportée par cette étude au projet général montre l'importance d'utiliser des outils automatiques pour mettre en avant des indices objectifs. Il n'en reste pas moins qu'il est nécessaire d'affiner les résultats. Notre perspective principale est de proposer une ressource normalisée, riche en méta-données (dont le manque et l'importance sont mis en valeur dans (Ghio *et al.*, 2006)), malgré les nombreuses difficultés éthiques que posent ces travaux.

Références

ASHER N. & LASCARIDES A. (2003). *Logics of Conversation*. Studies in Natural Language Processing. Cambridge University Press.

BARRAS C., GEOFFROIS E., WU Z. & LIBERMAN M. (1998). Transcriber : a free tool for segmenting, labeling and transcribing speech. In *International Conference on Language Resources and Evaluation (LREC)*, p. 1373–1376.

BENZITOUN C., FORT K. & SAGOT B. (2012). TCOF-POS : un corpus libre de français parlé annoté en morphosyntaxe. In *Traitement Automatique des Langues Naturelles (TALN)*, p. 99–112, Grenoble, France.

11. Les performances de ce outil avec ce modèle atteignent 97,61 % d'exactitude.

BLANCHE-BENVENISTE C. & JEANJEAN C. (1987). *Le Francais parlé. Transcription et édition*. Paris, France : Didier Érudition.

CHAIKA E. (1974). A linguist looks at "schizophrenic" language. *Brain and Language*, **1**(3), 257–276.

CONSTANT M. & DISTER A. (2010). Automatic detection of disfluencies in speech transcriptions. In I. C. F. D. M. PETTORINO, A. GIANNINI, Ed., *Spoken Communication*, volume 1, p. 259–272. Cambridge Scholars Publishing.

DE LA CLERGERIE É., SAGOT B., NICOLAS L. & GUÉNOT M.-L. (2009). FRMG : évolutions d'un analyseur syntaxique TAG du français. In É. VILLEMONTE DE LA CLERGERIE & P. PAROUBEK, Eds., *Journée de l'ATALA sur : Quels analyseurs syntaxiques pour le français ?*, Paris, France : ATALA. Journée de l'ATALA organisée conjointement à la conférence IWPT 2009.

DE MAREÜIL P. B., ADDA G., ADDA-DECKER M., BARRAS C., HABERT B. & PAROUBEK P. (2013). Une étude quantitative des marqueurs discursifs, disfluences et chevauchements de parole dans des interviews politiques. *TIPA. Travaux interdisciplinaires sur la parole et le langage [En ligne]*, **29**. mis en ligne le 19 décembre 2013, consulté le 14 février 2014.

DENIS P. & SAGOT B. (2009). Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art POS tagging with less human effort. In *Pacific Asia Conference on Language Information and Computing (PACLIC)*.

FELDSTEIN S. (1962). The relationship of interpersonal involvement and affectiveness of content to the verbal communication of schizophrenic patients. *Journal of Abnormal and Social Psychology*, **64**, 39–45.

FROMKIN V. A. (1975). A linguist looks at "a linguist looks at 'schizophrenic language"'. *Brain and Language*, **2**(0), 498 – 503.

GHIO A., TESTON B., VIALLET F., JANKOWSKI L., PURSON A., DUEZ D., LOCCO J., LEGOU T., PINTO S., MARCHAL A., GIOVANNI A., ROBERT D., RÉVIS J., FREDOUILLE C., BONASTRE J.-F., POUCHOULIN G. & NGUYEN N. (2006). Corpus de parole pathologique, état d'avancement et enjeux méthodologiques. *Travaux Interdisciplinaires du Laboratoire Parole et Langage d'Aix-en-Provence (TIPA)*, **25**, 109–126. Autorisation No.3015 : TIPA est la revue du Laboratoire Parole et Langage 3015 3015.

GOLDMAN-EISLER F., SKARBEK A. & HENDERSON A. (1965). The effect of chorpromazine on speech behaviour. *Psychopharmacologia*, **7**(3), 220–229.

KAMP H. & REYLE U. (1993). *From Discourse to Logic*. Kluwer Academic Publishers.

KREMEN W. S., SEIDMAN L. J., FARAONE S. V. & TSUANG M. T. (2003). Is there disproportionate impairment or phonemic fluency in schizophrenia ? *Journal of the International Neuropsychological Society*, **9**, 79–88.

LEVY R. (1968). The effect of chlorpromazine on sentence structure of schizophrenic patients. *Psychopharmacologia*, **13**(5), 426–432.

MAHER B. (1972). The language of schizophrenia : A review and interpretation. *The British Journal of Psychiatry*, **120**, 3–17.

MUSIOL M., AMBLARD M. & REBUSCHI M. (2013). Approche sémantico-formelle des troubles du discours : les conditions de la saisie de leurs aspects pyscholinguistiques. In *27ème Congrès International de Linguistique et de Philologie Romanes*, Nancy, France.

MUSIOL M. & TROGNON A. (1996). L'accomplissement interactionnel du trouble schizophrénique. *Raisons Pratiques 7*, p. 179–209.

PERRIER G. & GUILLAUME B. (2013). Leopar : an Interaction Grammar Parser. In *Workshop on High-level Methodologies for Grammar Engineering, ESSLLI*, p. 121–122, Dusseldorf.

REBUSCHI M., AMBLARD M. & MUSIOL M. (2013). Using SDRT to analyze pathological conversations. Logicality, rationality and pragmatic deviances. In M. REBUSCHI, M. BATT, G. HEINZMANN, F. LIHOREAU, M. MUSIOL & A. TROGNON, Eds., *Interdisciplinary Works in Logic, Epistemology, Psychology and Linguistics : Dialogue, Rationality, and Formalism*, Logic, Argumentation & Reasoning, p. 1–24. Springer.

URIELI A. & TANGUY L. (2013). L'apport du faisceau dans l'analyse syntaxique en dépendances par transitions : études de cas avec l'analyseur Talismane. In *Actes de la 20e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2013)*, p. 188–201, Les Sables d'Olonne, France.

VERHAEGEN F. (2007). *Psychopathologie cognitive des processus intentionnels schizophréniques dans l'interaction verbale*. PhD thesis, Université Nancy 2, France.

WARD D. (2008). *Stuttering and Cluttering : Frameworks for Understanding and Treatment*. Taylor & Francis.

Repérage et analyse de la reformulation paraphrastique dans les corpus oraux

Iris Eshkol-Taravella[1] Natalia Grabar[2]
(1) CNRS UMR 7270 LLL, Université d'Orléans, 45100 Orléans, France
iris.eshkol@univ-orleans.fr
(2) CNRS UMR 8163 STL, Université Lille 3, 59653 Villeneuve d'Ascq, France
natalia.grabar@univ-lille3.fr

Résumé. Notre travail porte sur la détection automatique de la reformulation paraphrastique dans les corpus oraux. L'approche proposée est une approche syntagmatique qui tient compte des marqueurs de reformulation paraphrastique et des spécificités de l'oral. L'annotation manuelle effectuée par deux annotateurs permet d'obtenir une description fine et multidimensionnelle des données de référence. Une méthode automatique est proposée afin de décider si les tours de parole comportent ou ne comportent pas des reformulations paraphrastiques. Les résultats obtenus montrent jusqu'à 66,4 % de précision. L'analyse de l'annotation manuelle indique qu'il existe peu de segments paraphrastiques avec des modifications morphologiques (flexion, dérivation ou composition) ou de segments qui montrent l'équivalence syntaxique.

Abstract. Our work addresses the automatic detection of paraphrastic rephrasing in spoken corpus. The proposed approach is syntagmatic. It is based on paraphrastic rephrasing markers and the specificities of the spoken language. Manual annotation performed by two annotators provides fine-grained and multi-dimensional description of the reference data. Automatic method is proposed in order to decide whether sentences contain or not the paraphrases. The obtained results show up to 66.4% precision. The analysis of the manual annotations indicates that there are few cases in which paraphrastic segments show morphological modifications (inflection, derivation or compounding) or syntactic equivalence.

Mots-clés : Paraphrase, reformulation, corpus oral, marqueurs de reformulation paraphrastique.

Keywords: Paraphrase, reformulation, spoken corpus, markers of paraphrastic rephrasing.

1 Introduction

La langue naturelle propose des moyens variés pour exprimer une même idée de différentes manières (exemples (1) à (8)). La paraphrase est liée à d'autres notions, comme la synonymie (Hamon *et al.*, 1998), la variation (Daille *et al.*, 1996; Grabar & Zweigenbaum, 2000), ou la reformulation paraphrastique (Roulet, 1987; Rossari, 1990; Bouamor *et al.*, 2012). Nous nous arrêterons sur les notions de paraphrase et de reformulation dans notre travail. L'acception de ces phénomènes varie selon les courants linguistiques : si chez les générativistes la notion de paraphrase n'est pas acceptable car toute modification de forme (*e.g.* nombre, mode, diathèse) implique un changement sémantique notable (Chomsky, 1975), cette notion connaît actuellement une acception très large notamment grâce aux travaux de TAL. Le critère commun, sur lequel tout le monde semble s'accorder aujourd'hui en parlant de la paraphrase, est qu'il existe entre les expressions linguistiques en relation de paraphrase une équivalence sémantique, qui peut prendre toutefois des formes différentes.

Parmi les fonctionnalités de la paraphrase se trouvent des phénomènes intra-locuteur et inter-locuteur. La paraphrase peut aider à une facilitation de la compréhension du message par l'autre et contribuer au bon fonctionnement de la communication (François, 1990; Bouamor *et al.*, 2012). Elle peut aussi empêcher la clarté de la communication (Boucheron, 2000) (*e.g.* la littérature scientifique spécialisée perçue par les non spécialistes). La paraphrase contribue à la beauté de la langue car elle évite les répétitions et redondances. Pour que la paraphrase soit détectée, la similitude entre l'information évoquée et l'information répétée doit être reconnue par l'interlocuteur. Du point de vue de TAL, la paraphrase crée une réelle difficulté : qu'il s'agisse de la reconnaissance ou de la production, la paraphrase couvre un nombre important de catégories, qui mettent en oeuvre des mécanismes linguistiques fort variés. Nous nous arrêterons sur les notions de paraphrase et de reformulation paraphrastique. Nous présentons d'abord les travaux en linguistique qui se sont attachés à décrire la paraphrase et la reformulation paraphrastique (section 1.1), et ensuite les travaux de TAL qui visent à détecter la paraphrase de manière automatique (section 1.2). Nous précisons ensuite les objectifs poursuivis dans notre travail (section 1.3).

1.1 Description linguistique de la paraphrase et de la reformulation paraphrastique

1.1.1 Typologies linguistiques de la paraphrase

Il existe différentes manières de traiter et de décrire la paraphrase. Par exemple, elle peut faire référence à la situation d'énonciation et avoir une valeur contextuelle. Ainsi, deux types de paraphrases sont distingués (Culioli, 1976; Martin, 1976; Vezin, 1976; Fuchs, 1994) :

– La paraphrase situationnelle est une "définition en discours" qui détermine le sens d'un énoncé (ou d'une partie d'énoncé) par rapport au contexte énonciatif. Dans l'exemple (1), *depuis une centaine d'années* et *depuis 1900* reçoivent ainsi une valeur contextuelle.

(1) *Depuis une centaine d'années, les températures du globe tendent à augmenter.*
Depuis 1900, les températures du globe tendent à augmenter.

– La paraphrase linguistique, parfois appelée une "définition en langue", est liée aux classifications linguistiques existantes. Nous en donnons quelques exemples dans ce qui suit.

Si l'on essaie de faire une typologie des transformations linguistiques subies par les entités, les niveaux suivants de la langue peuvent être distingués (Melčuk, 1988; Vila *et al.*, 2011; Bhagat & Hovy, 2013) :

– la paraphrase morphologique, qui a pour condition le changement morphologique (flexion, nominalisation, adjectivation, composition, etc.), comme dans l'exemple (2) :

(2) *Pierre a enlevé son manteau.* ; *Pierre enlève son manteau.*

– la paraphrase lexicale, qui vise le changement au niveau lexical (synonymes, antonymes, mots plus génériques ou spécifiques), comme dans l'exemple (3) :

(3) *Pierre a enlevé son manteau.* ; *Pierre a enlevé sa veste. Pierre a enlevé son vêtement.*

– la paraphrase sémantique, qui couvre en général des segments allant au-delà du lexique (exemple (4)) :

(4) *Pierre a enlevé son manteau.* ; *Pierre s'est déshabillé.*

– la paraphrase syntaxique, qui réorganise la phrase (déplacement de composants, diathèse...), comme dans l'exemple (5) ;

(5) *En entrant dans la bibliothèque, Pierre a enlevé son manteau.*
Pierre a enlevé son manteau en entrant dans la bibliothèque.

– la paraphrase mixte (*e.g.*, lexico-syntaxique, lexico-sémantique, etc.), qui concerne les modifications opérant simultanément à plusieurs niveaux (Bouamor *et al.*, 2012).

La notion de paraphrase peut aussi être décrite en fonction de la taille d'entités couvertes par la paraphrase (Flottum, 1995; Fujita, 2010; Bouamor, 2012) :

– la paraphrase lexicale, se situant au niveau d'un mot (exemples (6)) :

(6) {*bouquin*, *livre*}, {*bâtiment*, *maison*}

– la paraphrase sous-phrastique, avec laquelle la synonymie se fait au niveau des syntagmes, des tournures de phrases ou des fragments de textes (exemples (7)) :

(7) *Il a envie de*, *Il aimerait bien* ; *X ne doute pas de Y*, *X est sûr de Y*

– la paraphrase phrastique, où plusieurs segments sont en relation de paraphrase tandis que la sémantique de la phrase est préservée (exemples (8)) :

(8) *Comment vont vos enfants ?* ; *Comment se portent vos gamins ?* ; *Les gosses vont bien ?*

Les classifications existantes de la paraphrase focalisent souvent sur un aspect donné, décrit avec plus ou moins de finesse : 67 fonctions lexicales pour le paraphrasage (Melčuk, 1988), 25 catégories de paraphrases (Bhagat & Hovy, 2013). À notre connaissance, la seule classification multidimensionnelle est celle de (Milicevic, 2007), avec les dimensions suivantes :

– type de connaissances mis en jeu pour la production de paraphrases,
– modifications de sens impliquées,
– types de moyens d'expression utilisés (cette dimension est proche d'autres classifications existantes),

– exactitude du lien paraphrastique,
– mode de production.

Notons que la notion de paraphrase peut également couvrir d'autres dimensions :
– registre de langue (les équivalences inter-discours (Elhadad & Sutaria, 2007; Deléger & Zweigenbaum, 2008), les niveaux soutenu ou parlé de la langue, etc.),
– la langue (les équivalences inter-langues ou les traductions (Fuchs, 1982; Milicevic, 2007)).

Nous parlons de la paraphrase au sens large, tout en la réservant aux expressions d'une seule langue. Nous considérons ainsi que la paraphrase peut être utilisée non seulement pour reformuler mais aussi pour décrire, exemplifier, préciser ou expliquer une idée exprimée auparavant par un locuteur.

1.1.2 Reformulation paraphrastique à l'oral

La reformulation est propre autant à la langue soutenue, comme celle des articles scientifiques, qu'à la langue parlée, bien qu'elle montre des différences dans les deux cas (Flottum, 1995; Rossari, 1992). Ainsi, dans l'écrit, c'est le produit fini qui se présente au destinataire (Hagège, 1985), alors que l'oral l'exhibe dans les étapes de son élaboration. Il est en effet commun de trouver dans la langue orale des traces de sa propre production (*e.g.* hésitations, faux-départs, formes diverses de reprises) à la manière de brouillons qui précèdent la version finale des écrits (Blanche-Benveniste *et al.*, 1991). De manière générale, il est considéré que la reformulation est une activité du locuteur qui s'appuie sur un segment déjà produit dans son propre discours ou dans celui de son interlocuteur, avec ou sans l'emploi d'un marqueur, afin d'en modifier certains aspects (lexical, syntaxique, sémantique, pragmatique) tout en gardant un invariant permettant de reconnaître l'opération ainsi mise en place (Gülich & Kotschi, 1987; Kanaan, 2011). Pour ces différentes raisons, tout acte de reformulation dans le discours oral n'introduit pas toujours une paraphrase (Rossari, 1990). De ce point de vue, on distingue deux catégories de marqueurs : les marqueurs de reformulation non-paraphrastique (*e.g. en somme, en tout cas, de toute façon, enfin*, etc.) et les marqueurs de reformulation paraphrastique (ou MRP), comme *c'est-à-dire, autrement dit, je m'explique, ça veut dire, en d'autres termes* (Rossari, 1990, 1993). Les critères, qui permettent de détecter la reformulation paraphrastique sont (Gulich & Kotschi, 1983; Rossari, 1993) :
– trois critères phonétiques : répétition du contour intonatif de la phrase ; réduction de la vitesse de débit ; et articulation remarquablement nette des deux syllabes qui terminent l'énoncé doublon ;
– parallélisme syntaxique entre l'entité source et l'entité paraphrasée ;
– présence d'un MRP, bien qu'il soit possible d'avoir une relation de paraphrase sans marqueur. Parmi les MRP, les auteurs distinguent ceux qui ont pour tâche principale d'établir une relation paraphrastique (*e.g. c'est-à-dire, autrement dit*) et ceux qui ne montrent ce rôle que dans des contextes précis.

Les MRP fournissent un marquage formel de liens paraphrastiques entre deux segments : segment source et segment cible (ou paraphrasé). Les propriétés sémantiques des MRP permettent d'instaurer une relation de paraphrase même entre les segments qui n'entretiennent aucune équivalence sémantique visible par ailleurs (Rossari, 1993).

1.2 Description et détection de la paraphrase en TAL

Deux états de l'art récents sur les méthodes pour la détection automatique de la paraphrase (Madnani & Dorr, 2010; Androutsopoulos & Malakasiotis, 2010) montrent l'intérêt important réservé à ces méthodes et ressources dans le domaine du TAL. Les approches proposées pour la détection automatique de la paraphrase dépendent du type de corpus exploités et reposent généralement sur les propriétés paradigmatiques des mots (leur capacité de se substituer mutuellement) :

1. *Corpus monolingues.* En corpus monolingues, la similarité des chaînes d'édition (Malakasiotis & Androutsopoulos, 2007) et les méthodes distributionnelles sont le plus souvent utilisées. Dans ce dernier cas, si les unités linguistiques (mots, syntagmes, etc) ont des vecteurs similaires, elles sont alors de bons candidats pour la paraphrase (Lin & Pantel, 2001; Pasça & Dienes, 2005) ;
2. *Corpus monolingues parallèles.* Lorsqu'un texte dans une langue est traduit plus d'une fois dans une autre langue, les traductions de ce texte permettent de constituer un corpus monolingue parallèle. Un des plus utilisés est constitué des traductions en anglais de *20 000 lieux sous la mer* de Jules Verne. L'exploitation de tels corpus est notamment possible grâce aux méthodes d'alignement de mots (Och & Ney, 2000). Différentes méthodes ont été proposées pour l'exploitation de tels corpus (Barzilay & McKeown, 2001; Ibrahim *et al.*, 2003; Quirk *et al.*, 2004) ;
3. *Corpus monolingues comparables.* Les corpus monolingues comparables contiennent typiquement des textes produits indépendamment sur un même événement, comme par exemple les articles de presse qui couvrent l'actualité. La cohérence thématique de ces textes d'un côté et les méthodes distributionnelles ou bien l'alignement de phrases

comparables de l'autre côté permettent d'induire les relations de paraphrase entre les segments de texte (Shinyama *et al.*, 2002; Sekine, 2005; Shen *et al.*, 2006) ;

4. *Corpus bilingues parallèles.* Les corpus bilingues parallèles, qui contiennent typiquement la traduction d'un texte dans une autre langue, peuvent aussi être utilisés pour la détection de la paraphrase. Dans cette situation, les traductions multiples d'une expression ou d'un mot peuvent correspondre aux paraphrases (Bannard & Callison-Burch, 2005; Madnani *et al.*, 2008; Callison-Burch *et al.*, 2008; Kok & Brockett, 2010).

1.3 Objectifs

L'objectif que nous poursuivons dans notre travail concerne la détection de reformulations paraphrastiques. L'originalité du travail proposé consiste en points suivants :
– Le corpus de travail est un corpus oral, très peu exploité jusqu'ici pour détecter les paraphrases (Bouamor *et al.*, 2012) ;
– La méthode choisie pour détecter les reformulations paraphrastiques dans un corpus monolingue est une approche syntagmatique et non distributionnelle réservé à ce type de matériel (Madnani & Dorr, 2010) ;
– Une annotation multidimensionnelle de la paraphrase est proposée. Elle permet de créer les données de référence ;
– La distinction automatique entre les reformulations paraphrastiques et non-paraphrastiques est effectuée.
Nous décrivons d'abord les données exploitées (section 2) et les méthodes proposées (section 3). Nous présentons et discutons ensuite les résultats dans la section 4, et terminons avec des perspectives de recherches (section 5).

2 Données linguistiques

2.1 Corpus

Nous travaillons avec les corpus ESLO (Enquêtes Sociolinguistiques à Orléans) (Eshkol-Taravella *et al.*, 2012) : *ESLO1* et *ESLO2*. *ESLO1*, la première enquête sociolinguistique à Orléans, a été réalisée en 1968-1971 par des professeurs de français de l'University of Essex, Language Centre, Colchester (Royaume-Uni), en collaboration avec des membres du B.E.L.C. (Bureau pour l'étude de l'enseignement de la langue et de la civilisation françaises de Paris). Le corpus *ESLO1*, constitué à Orléans mais archivé ensuite de manière fragmentaire ailleurs, est revenu dans les années 1990 au LLL (Laboratoire Ligérien de Linguistique). Le laboratoire a mis au format standard ce corpus d'enquêtes sociolinguistiques comprenant 300 heures de parole (4 500 000 mots environ) incluant une gamme d'enregistrements variés. En prenant en compte l'expérience d'*ESLO1* et l'évolution des cadres théoriques et méthodologiques de la constitution et de l'exploitation de grands corpus oraux à visée variationniste, une nouvelle enquête *ESLO2* a été entamée en 2008. À terme, *ESLO2* comprendra plus de 350 heures d'enregistrements afin de former avec *ESLO1* un corpus de plus de 700 heures et d'atteindre les dix millions de mots. Les corpus *ESLO1* et *ESLO2* sont accessibles en ligne (*http ://eslo.tge-adonis.fr/*).

Pour avoir des données comparables dans les deux corpus, nous avons sélectionné 260 entretiens d'*ESLO1* totalisant 2 349 829 occurrences de mots et 308 entretiens d'*ESLO2* totalisant 1 412 891 occurrences de mots. Les fichiers transcrits d'ESLO respectent deux principes : l'adoption de l'orthographe standard et le non-recours à la ponctuation de l'écrit. La segmentation est faite soit sur une unité intuitive de type "groupe de souffle" repérée par le transcripteur humain, soit sur un *tour de parole*, défini uniquement par le changement de locuteurs. Nous avons utilisé les versions C de transcription. Ces fichiers de transcription n'ont pas été corrigés et ont été pris comme tels.

2.2 Marqueurs de reformulation paraphrastique (MRP)

Nous exploitons trois MRP : *c'est-à-dire*, *je veux dire* et *disons*. Le point commun entre eux est qu'ils sont formés à partir du même verbe *dire*. Le marqueur *c'est-à-dire* est le plus lexicalisé des trois et semble être le plus étudié. Les propriétés qu'on lui reconnaît sont les suivantes (Gulich & Kotschi, 1983; Hölker, 1988; Beeching, 2007) :
– il est utilisé dans les monologues et dans les dialogues, à l'écrit et à l'oral ;
– les éléments liés ne peuvent pas être échangés car ils n'ont pas d'égalité entre eux ;
– *c'est-à-dire* peut commuter avec *à savoir, en réalité, autrement dit, en d'autres termes* et *donc* ;
– ce marqueur peut instaurer la relation de paraphrase entre des énoncés non équivalents sémantiquement ;
– les trois fonctions prototypiques de ce marqueur sont : corrigeante, reformulante et argumentative, mais il peut aussi marquer la conclusion (il est alors substituable par *donc*), la justification ou l'hésitation.

Les caractéristiques du marqueur *disons* ont été montrées dans (Hwang, 1993) :
– il est assez proche de *je dirais, je veux dire* ;
– il existe une analogie entre *eh bien* et *disons* du point de vue énonciatif, car ils marquent tous les deux une rupture : en mettant fin au niveau coénonciatif précédent, le locuteur signale l'ouverture d'un plan énonciatif différent et égocentré ;
– il existe une analogie entre *disons* et *enfin* en tant que moyens de rectification.
(Saunier, 2012) distingue six pôles pour décrire les différents sens de *disons* : *à peu près*, *en bref*, *ou plutôt*, *plus précisément*, *oui et non* et *ni la chèvre ni le chou*. Pour (Petit, 2009), il est impossible de supprimer *disons* de l'énoncé parce que le deuxième segment exprime une nuance sémantique différente. Souvent, il est lié à la recherche d'un terme plus adéquat. En ce qui concerne le marqueur *je veux dire*, (Teston-Bonnard, 2008) identifie plusieurs *je veux dire* et démontre ainsi que deux des statuts syntaxiques repérés (verbes recteurs faibles et parenthèses) se paraphrasent par *autrement dit, c'est-à-dire, je reprends*. Comme le montrent les recherches citées ci-dessus, ces trois éléments peuvent avoir des fonctions différentes. Dans notre travail, nous nous intéressons à eux en tant que marqueurs de reformulation paraphrastique.

3 Méthodologie pour la détection de paraphrases

Les tours de parole avec les trois MRP sont extraits des deux corpus et pré-traités (section 3.1). La méthode proposée est fondée sur le traitement manuel (section 3.2) et automatique (section 3.3) de ces corpus. Nous effectuons également une analyse et évaluation des résultats (section 3.4).

3.1 Préparation des corpus

Une des plus grandes difficultés du traitement automatique des transcriptions de l'oral est l'absence de marques formelles de segmentation. Si à l'écrit les signes de ponctuation remplissent bien cette fonction, à l'oral ce sont les éléments paralinguistiques qui marquent le début et la fin de l'énoncé : la pause, l'intonation, etc. Pour résoudre en partie ce problème, nous avons utilisé comme segmenteur un tour de parole marqué dans la transcription par un changement de locuteur. La difficulté s'est posée alors pour les cas de chevauchement où les deux locuteurs parlent en même temps. Dans ces situations, les segments correspondants sont associés aux énoncés de chacun des locuteurs impliqués et lorsqu'un locuteur continue de parler après un chevauchement, son tour de parole continue. Les corpus sont ensuite traités avec le chunker SEM (Eshkol *et al.*, 2014) adapté à la langue orale. SEM détecte les chunks minimaux, comme présenté dans l'exemple (9) (même exemple qu'en (12)).

(9) *(est/V)VN (-ce/CLS)NP (que/CS)CONJ (vous/CLS)NP (remarquez/V)VN (une/DET différence/NC sensible/ADJ)NP (entre/P vos/DET différents/ADJ clients/NC dans/P leur/DET façon/NC de/P choisir/VINF)PP (la/DET viande/NC) NP (dans/P ce/PRO)PP (qu'/PROREL ils/CLS)NP (achètent/V)VN (et/CC)CONJ (caetera/V)VN (,/CLS)NP (indépendamment/V disons/VPP)VN (de/P leurs/NC)PP (oui/I)IntP (origines/NC)NP (de/P classe/NC ,/ADJ)PP*

3.2 Annotation manuelle des reformulations paraphrastiques

L'annotation manuelle a pour objectif de distinguer entre les reformulations paraphrastiques et les reformulations non paraphrastiques, mais aussi de proposer une annotation plus fine. Pour les contextes de reformulation paraphrastique, l'annotation est focalisée sur les deux segments (des mots ou des segments plus grands) mis en relation de reformulation autour d'un MRP, mais aussi sur la relation établie par le MRP de manière générale. L'annotation est effectuée en suivant plusieurs dimensions, dont certaines sont inspirées par les classifications existantes (section 1.1.1) :

1. *catégorie syntaxique* : les segments mis en relation sont annotés par leur catégorie syntaxique (N, A, V, Prep...) ou leur type de constituant syntaxique (NP, VP, AP, PP). Cela permet également de voir s'il existe une équivalence syntaxique entre les deux segments. Les segments mis en relation ne sont pas définis sur une base syntaxique (*e.g.* des chunks), mais sur le critère sémantique de paraphrase qui se trouve à la base de cette relation ;
2. chaque relation est annotée avec plusieurs arguments évocateurs des classifications existantes de la paraphrase (section 1.1.1). S'il existe plusieurs éléments paraphrasés, ils sont tous annotés de cette manière :
   – *rel-lex* : type de la relation lexicale entre deux éléments paraphrasés : hyperonyme, synonyme, antonyme, instance, méronyme ;
   – *modif-lex* : type de la modification lexicale : remplacement, suppression, ajout ;

– *modif-morph* : type de la modification morphologique : flexion, dérivation, composition ;
– *modif-synt* : type de la modification syntaxique : passif/actif... ;

3. *rel-pragm* : type de la relation pragmatique. Cette relation est liée aux fonctionnalités de la paraphrase ou de la reformulation. Notre typologie est inspirée des typologies proposées dans la littérature (Gülich & Kotschi, 1987; Kanaan, 2011). Parmi ces fonctionnalités, nous distinguons : définition, explication, exemplification, précision, dénomination, résultat, correction linguistique, correction référentielle, équivalence.

Les exemples (10) et (11) montrent le résultat de cette annotation (les annotations sont en bleu, les références des fichiers *ESLO* entre les crochets). Nous pouvons ainsi voir que les segments {*Saint Jean de la Ruelle*, *Orléans*} (exemple (10)) et {*démocratiser l'enseignement*, *permettre à tout le monde de rentrer en faculté*} (exemple (11)) sont en relation de paraphrase, tout en mettant en jeu des mécanismes linguistiques et pragmatiques différents.

(10) *pendant nous avons fait grève à la Régie Renault euh de <NP1>Saint Jean de la Ruelle</NP1> <MRP>c'est-à-dire</MRP> <NP2 rel-lex="mer(Saint Jean de la Ruelle/Orléans)" rel-pragm="cor-ref">Orléans</NP2> parce que c' est ça fait partie d' Orléans* [ESLO1_ENT_149_C]

(11) *euh <VP1>démocratiser l'enseignement</VP1> <MRP>c'est-à-dire</MRP> <VP2 rel-lex="syn(démocratiser/permettre à tout le monde) syn(enseignement/faculté)" modif-lex="ajout(rentrer à)" rel-pragm="explic">permettre à tout le monde de rentrer en faculté</VP2>* [ESLO1_ENT_121_C]

3.3 Détection automatique de reformulations paraphrastiques

Le traitement automatique principal consiste à décider si, autour d'un MRP, il existe une relation de reformulation paraphrastique ou non. Pour ceci, nous mettons en place plusieurs filtres qui sont communs à tous les MRP :
– Si le MRP est placé en début ou en fin de TdP, alors il est considéré que ce TdP ne comporte pas de paraphrase ;
– Si le MRP est entouré des marqueurs discursifs (*donc, enfin, quoi...*), euh d'hésitation, interjections (*ben hm ouais*), amorces (*s-*), etc. répétés, il est considéré que le MRP fait partie des disfluences de l'oral (une accumulation d'éléments qui brisent le déroulement syntagmatique (Blanche-Benveniste *et al.*, 1991)) et n'introduit pas la paraphrase ;
– Si le MRP apparaît dans un contexte lexical spécifique (emploi de *nous* devant *disons*), ou si le MRP apparaît dans des suites argumentatives (*e.g. par contre, mais, en revanche, au contraire*), ce TdP ne comporte pas de paraphrase ;
– Si le MRP apparaît à l'intérieur d'une locution, comme *indépendamment de* ou *plus ou moins grossiers* (exemples (12) et (13) de la section 4.2), alors il est considéré que ce contexte ne comporte pas de paraphrase. Ce test est effectué sur les sorties du chunker (exemple (9)). Pour vérifier que la locution existe dans la langue, nous interrogeons un moteur de recherche généraliste et analysons les fréquences attestées sur la Toile. À notre avis, l'usage de la Toile fournit des informations plus complètes que celles que l'on peut trouver dans des corpus de référence. Chaque segment est testé de trois manières (exemple (9)) : avec un (*(caetera)VN (indépendamment)VN (de leurs)PP*), deux (*(et)CONJ (caetera)VN (indépendamment)VN (de leurs)PP (origines)NP*) ou trois chunks *(achètent)VN (caetera)VN (indépendamment)VN (de leurs)PP (origines)NP (de classe)PP* à droite et à gauche du MRP, excepté les disfluences et la ponctuation. La taille maximale du segment est empiriquement limitée à sept mots. La fréquence moyenne de ces segments doit être inférieure au seuil entre 10 et 6 000 pour que ce segment soit considéré comme une paraphrase. Dans le cas de fréquences plus élevées que le seuil, ce test indique que la locution existe dans la langue et qu'il s'agit en effet de disfluence.

3.4 Analyse et évaluation

L'évaluation est effectuée de deux manières :
– pour l'annotation manuelle, nous calculons l'accord inter-annotateur pour les jugements sur l'existence de la relation de paraphrase. Comme deux annotateurs ont participé à cette tâche, nous appliquons le kappa de Cohen (Cohen, 1960). Le protocole d'annotation a été mis en place et ajusté sur une partie du corpus *ESLO1*, tandis que l'évaluation et l'accord inter-annotateur sont calculés sur d'autres TdP du corpus *ESLO1* et sur la partie *entretiens* du corpus *ESLO2* ;
– pour la détection automatique de relations paraphrastiques, elle est évaluée par rapport aux annotations manuelles. Nous calculons la précision des résultats.

L'analyse des résultats porte sur une étude de la fréquence des relations et de la répartition des différents attributs de manière générale et en fonction des MRP. L'accent principal est mis sur l'existence de relations paraphrastiques, mais aussi sur l'équivalence syntaxique entre les segments mis en relation et sur l'existence de modifications morphologiques (flexion, dérivation ou composition), qui peuvent donner des indications formelles de paraphrasage.

4 Résultats et Discussion

4.1 Préparation des corpus

	ESLO1	*ESLO2*
nombre de fichiers de transcription	260	308
taille de corpus (occ de mots)	2 349 829	1 412 891
taille moyenne des fichiers de transcription	9 037,80	4 587,31
nombre de TdP	166 602	70 707
taille moyenne des TdP	14,10	19,98
c'est-à-dire	1 849	594
je veux dire	285	291
disons	1 068	183
total de TdP avec les MRP	3 202	1 068
taille des TdP avec les MRP (minimale)	1	1
taille des TdP avec les MRP (maximale)	6 382	1 050
taille des TdP avec les MRP (moyenne)	62,88	86,34

TABLE 1 – Différentes informations sur les données : taille des corpus, nombre et taille moyenne des TdP, nombre de TdP avec les trois MRP étudiés, taille des TdP avec les MRP.

Le tableau 1 présente la taille des corpus en nombre de mots et indique différentes informations sur les extractions effectuées : nombre et taille moyenne des TdP, nombre de TdP avec les trois MRP étudiés, taille des TdP avec les MRP. Ces chiffres montrent que la taille moyenne des TdP est entre 14 et 19 mots : de nombreux TdP sont en effet de taille minimale, avec un ou deux mots seulement. Le marqueur *c'est-à-dire* est toujours le plus fréquent, avec plus de la moitié des TdP contenant un MRP. Rappelons que *c'est-à-dire* établit principalement une relation paraphrastique même entre les TdP ayant une équivalence sémantique faible (Gulich & Kotschi, 1983). Quant à *disons*, il est très fréquent dans le corpus *ESLO1* mais beaucoup moins dans *ESLO2*. Il est possible que ce soit dû à l'évolution diachronique de la langue : d'autres mots ont pu reprendre cette fonction discursive. En ce qui concerne la taille moyenne des TdP avec les MRP, elle est assez élevée (62,88 dans *ESLO1* et 86,34 dans *ESLO2*). Ces TdP peuvent en effet comporter des paraphrases et montrer la genèse et la précision des idées (Hagège, 1985; Blanche-Benveniste *et al.*, 1991) de la part des locuteurs. Nous pouvons aussi observer que la taille maximale des TdP peut aller jusqu'à 1 050 dans *ESLO2* et 6 382 dans *ESLO1*.

4.2 Annotation manuelle des paraphrases

	ESLO1					*ESLO2*				
	A1		*A2*		*accord*	*A1*		*A2*		*accord*
	oui (%)	*non (%)*	*oui (%)*	*non (%)*		*oui (%)*	*non (%)*	*oui (%)*	*non (%)*	
c'est-à-dire	96 (33)	193 (67)	66 (23)	223 (77)	249	74 (37)	124 (63)	65 (32)	137 (68)	162
je veux dire	16 (25)	49 (75)	8 (12)	57 (88)	57	47 (34)	91 (66)	27 (20)	110 (80)	107
disons	18 (15)	104 (85)	8 (7)	115 (93)	106	10 (18)	45 (82)	9 (16)	46 (84)	46
total de TdP	130 (27)	346 (73)	82 (17)	395 (83)	412	131 (33)	260 (67)	101 (26)	293 (74)	315

TABLE 2 – Jugements sur la relation de paraphrase dans les contextes avec les MRP : pour deux annotateurs et leur accord.

Tout MRP confondu, 476 TdP du corpus *ESLO1* et 394 TdP du corpus *ESLO2* de la partie *entretiens* sont analysés (54 et 30 entretiens respectivement). Cette annotation permet de proposer un premier jeu de données de référence et un guide d'annotation. Le tableau 2 montre les résultats des annotations par les deux annotateurs impliqués. Les annotateurs reconnaissent entre 17 et 27 % de contextes paraphrastiques dans le corpus *ESLO1* et entre 26 et 33 % de contextes paraphrastiques dans le corpus *ESLO2*. L'annotateur *A1* a tendance à accepter plus de contextes comme paraphrastiques, ce qui montre l'aspect subjectif de ce type d'annotation. La perception de paraphrase varie en effet d'un annotateur à l'autre. L'accord entre les annotateurs est de 0,617 pour *ESLO1*, ce qui correspond à un accord substantiel, et de 0,526

pour *ESLO2*, ce qui correspond à un accord modéré (Landis & Koch, 1977). Il s'agit d'un niveau d'accord assez important, surtout lorsque l'on travaille avec des données linguistiques qui peuvent introduire la subjectivité dans leur perception. Comme annoncé dans la littérature, ces MRP peuvent apparaître dans des emplois paraphrastiques et non-paraphrastiques. Dans les exemples (12) et (13), qui ne contiennent pas de relations de paraphrase, les MRP peuvent ainsi être associés aux marqueurs discursifs faisant partie des disfluences.

(12) *est-ce que vous remarquez une différence sensible entre vos différents clients dans leur façon de choisir la viande dans ce qu'ils achètent et caetera indépendamment <MRP>disons</MRP> de leurs oui origines de classe* [ESLO1_ENT_001_C]

(13) *mais il y a des termes qui sont plus ou moins euh euh <MRP>disons</MRP> euh grossiers qui sont employés plus ou- plus facilement euh par certaines couches de la société en fonction des fréquentations des uns ou des autres quoi* [ESLO1_ENT_003_C]

	ESLO1				*ESLO2*			
	A1		*A2*		*A1*		*A2*	
	oui	*non*	*oui*	*non*	*oui*	*non*	*oui*	*non*
c'est-à-dire (%)	33	67	22	78	37	63	32	68
je veux dire (%)	25	75	12	88	34	66	20	80
disons (%)	15	85	7	93	18	82	6	94

TABLE 3 – Pourcentage des constructions paraphrastiques autour des MRP.

Le tableau 3 indique le pourcentage des constructions paraphrastiques et non-paraphrastiques autour des MRP. Pour les deux annotateurs, *c'est-à-dire* est le plus grammaticalisé de ce point de vue car il introduit le plus de relations de paraphrase, tandis que *disons* est le moins grammaticalisé. *je veux dire*, qui a la position intermédiaire, est plus proche de *c'est-à-dire*. Concernant *disons*, nous pensons qu'il est le plus ambigu des trois MRP car d'une part il peut signifier le verbe *dire* et donc, en quelque sorte, signifier l'emploi contraire à la paraphrase où il marque le début d'une nouvelle idée : le locuteur introduit alors quelque chose de nouveau comme dans *disons que...* D'autre part il peut être employé en tant que marqueur discursif, associé aux *expressions stéréotypées* fonctionnant en tant que adverbe, conjonction, interjection, etc. (Gulich & Kotschi, 1983), ou disfluences (exemples (12) et (13)).

Dans la majorité des cas (plus de 70 %), il n'existe pas d'équivalence syntaxique entre les éléments en relation de paraphrase (comme dans les exemples (14) et et (15)). Notons que cet aspect dépend du choix des annotateurs. Ainsi, dans l'exemple (14), au lieu de la proposition *les gens me semblent plus plus affables* il est aussi possible de choisir le syntagme adjectival *plus affables* ou l'adjectif *affables*. Un autre aspect intéressant de l'annotation est lié aux modifications morphologiques observables entre les segments en relation de paraphrase. Nous pouvons ainsi voir que de telles modifications sont annotées pour environ dix relations paraphrastiques par corpus, tout MRP confondus. En voilà quelques exemples : {*achat*, *achète*}, {*connais*, *connu*}, {*pourrait*, *pouviez*}, {*client*, *clientèle*}, {*manoeuvres*, *manuel*}, {*aller*, *vais*}. Cela indique qu'il existe très peu d'accroches formelles pour détecter les segments en relation de paraphrase dans ce type de constructions. Les modifications syntaxiques, comme par exemple le changement de la voix active en voix passive, ne sont quasiment pas présentes, avec seulement un exemple au sein des 54 et 30 entretiens annotés respectivement dans *ESLO1* et *ESLO2*. En ce qui concerne les modifications lexicales, nous observons surtout le remplacement d'un sous-segment par un autre. Comme cela a été noté dans la littérature (Gulich & Kotschi, 1983; Rossari, 1993), dans plusieurs cas, nous rencontrons effectivement des segments, qui n'ont aucun lien sémantique évident, mais, grâce à un MRP et à la relation de paraphrase établie, ce lien peut apparaître (exemples en (16) ou (17)).

(14) *je préfère mieux le le nord de la France franchement le département du Nord et le département du Pas-de-Calais où <P1>les gens me semblent plus plus affables</P1> <MRP>disons</MRP> euh <PP2 rel-lex="syn" rel-pragm="explic">avec qui j' ai on a plus facilement des des rapports agréables</PP2>* [ESLO1_ENT_003_C]

(15) *y a le euh le le plus grand goup- groupe et puis euh ce qu'on appelle <NP1>toujours les mêmes</NP1> <MRP>c'est-à-dire</MRP> euh <P2 rel-lex="syn" rel-pragm="equiv">tous ceux qu'on connait</P2> quoi* [ESLO2_ENT_1004_C]

(16) *des conférences y en a assez souvent sur France culture enfin <MRP>disons</MRP> des causeries* [ESLO1-

_ENT_121_C]

Parmi les relations lexicales, les plus fréquentes sont les relations de synonymie et d'hypéronymie, suivies par les instances dans le cas des entités nommées, et l'équivalence et le résultat. En fonction des relations pragmatiques assignées, nous pouvons distinguer trois fonctions effectuées par les MRP (dans l'ordre décroissant de fréquence dans les corpus) :
– la possibilité d'ajouter une nouvelle information, notée par des relations pragmatiques d'explication, de précision, d'exemplification et de définition. Cette fonction peut être rapprochée des fonctions corrigeante, reformulante et argumentative notées dans la littérature (Gulich & Kotschi, 1983; Hölker, 1988). Dans tous ces cas, il s'agit de rendre l'énoncé plus riche et clair, comme dans les exemples (14) et (11) ;
– l'établissement de la relation d'équivalence : redire la même chose, mais avec d'autres moyens linguistiques. Ce qui est intéressant avec l'équivalence, mais aussi avec la définition, est que, contrairement à ce qui a été observé dans la littérature (Gulich & Kotschi, 1983; Petit, 2009), il est possible de supprimer le MRP et d'échanger les segments de place sans que cela modifie la sémantique de l'énoncé (exemples (15) et (16)) ;
– avec la relation *résultat*, nous pouvons observer le phénomène inverse à l'explication : le deuxième segment peut être réduit par rapport au premier et en proposer une synthèse (exemple (17)).

(17) *voilà <P1>le côté très betonné voilà c'est pas ils ont pas développé les les logements étudiants suffisamment ils ont pas développé l'off- l'offre culturelle euh en même temps</P1> donc enfin <MRP>je veux dire</MRP> voilà <P2 rel-pragm="res">c'est mort</P2>* [ESLO2_ENT_1012_C]

4.3 Détection automatique de reformulations paraphrastiques

	ESLO1		*ESLO2*	
	A1	A2	A1	A2
filtres	40,5	40,5	37,7	37,8
filtres + fréquences (>6000)	25,8	25,9	18,7	18,9
filtres + fréquences prioritaires (>6000)	63,0	63,0	66,4	66,3

TABLE 4 – Évaluation de la détection automatique des reformulations paraphrastiques (précision).

Les résultats sur la détection automatique des reformulations paraphrastiques se trouvent dans le tableau 4. L'évaluation est effectuée en termes de précision. Notons que les résultats sont cohérents entre les deux annotateurs dans les deux corpus. Quant au jugement sur la présence de reformulations paraphrastiques, nous observons que les filtres prenant en charge les disfluences orales permettent d'atteindre jusqu'à 40 % de précision. L'ajout de filtres supplémentaires (fréquences sur la Toile) aux filtres de disfluences détériore les résultats : nous pouvons perdre jusqu'à 18 %. Par contre, lorsque les fréquences sont prioritaires sur les filtres de disfluences, les résultats sont améliorés et peuvent atteindre jusqu'à 66,4 % de précision. Dans cette configuration, si les fréquences satisfont nos critères, nous considérons qu'un TdP peut contenir une paraphrase même si le MRP se trouve dans un contexte de disfluences de l'oral. Les résultats s'améliorent avec l'augmentation du seuil. La seuil maximum testé est 6 000, l'augmentation est observée jusqu'au seuil 4 500.

Nos recherches montrent également qu'il existe des schémas plus compliqués que ceux décrits dans les travaux antérieurs (Rossari, 1990) et ceux pris actuellement en charge par notre système :
– la relation de paraphrase peut aussi se construire sur plus d'un TdP : lorsque le locuteur est interrompu sans chevauchements et lorsqu'il continue son discours plus loin. Actuellement, nous ne traitons pas ce type de situations car la détection de paraphrases est effectuée au sein d'un même TdP ;
– nous avons distingué deux situations selon que les segments en relation de la paraphrase sont contigus ou distants par rapport au MRP. Ainsi, dans l'exemple (18), les segments sont distants. Comme noté, ils peuvent être séparés du MRP par des disfluences ou bien par d'autres segments à contenu. Nous avons essayé de prendre en compte la possibilité d'avoir des disfluences intercalées, ce qui améliore en effet les résultats globaux ;
– le MRP peut aussi se trouver non pas entre mais après les deux segments en relation de paraphrase (exemple en (19)).

(18) *<PP1>jusqu'à seize ans</PP1> oui oui bien bon <MRP>c'est-à-dire</MRP> euh <PP2 rel-lex="syn" rel-pragm="cor-ref">dans le primaire privé</PP2> n'est-ce pas* [ESLO1_ENT_010_C]

(19) *et elle travaille euh dans des instituts d' enfants euh plus ou moins adaptés enfin plutôt inadaptés <MRP>disons </MRP> et en plus de ça elle a une clientèle personnelle ici* [ESLO1_ENT_003_C]

5 Conclusion et perspectives

Nous avons proposé un travail sur la détection de reformulations paraphrastiques dans des corpus monolingues oraux du français (corpus *ESLO1* et *ESLO2*). Une des originalités du travail consiste à prendre en compte les spécificités de l'oral, que ce soit grâce à la reconstitution des tours de parole, à la considération des disfluences dans le contexte ou à l'utilisation d'outils de TAL adaptés à l'oral (Eshkol *et al.*, 2014). Un autre aspect original est que nous abordons la détection de reformulations paraphrastiques avec une approche syntagmatique, alors que la plupart des approches existantes pour la détection de paraphrases exploitent les propriétés paradigmatiques de la langue. Notre travail repose sur une utilisation combinée de l'annotation manuelle et d'un traitement automatique des données. L'annotation permet de produire un premier jeu de données de référence et de faire plusieurs observations sur les relations de paraphrase, en particulier grâce à une annotation multidimensionnelle et fine. Elle a servi aussi à l'évaluation de la méthode automatique et peut être exploitée dans l'apprentissage automatique. L'accord inter-annotateur obtenu est de 0,617 et 0,526 pour les corpus *ESLO1* et *ESLO2* respectivement. Rappelons aussi que nous adoptons une notion large de la paraphrase (Melčuk, 1988; Bhagat & Hovy, 2013), qui peut clarifier, expliciter, développer ou résumer l'idée exprimée auparavant par un locuteur. Les traitements automatiques proposent une série de critères pour distinguer entre les contextes paraphrastiques et non-paraphrastiques. La précision obtenue est 66,4 %. Si ces critères sont élaborés et testés sur des corpus oraux et au sein d'une approche syntagmatique, nous pensons qu'ils peuvent être transposables à d'autres corpus.

Par rapport aux travaux existants, l'analyse des corpus proposée ici confirme les constatations faites par les chercheurs :
– la reformulation n'est pas toujours paraphrastique et les MRP n'introduisent pas toujours des relations de paraphrase (Rossari, 1990), les MRP pouvant effectivement assumer d'autres rôles dans la langue ;
– les MRP peuvent instaurer la relation de paraphrase entre les segments qui n'entretiennent aucune équivalence sémantique visible (Gulich & Kotschi, 1983; Rossari, 1993).
Pour d'autres constatations faites dans la littérature (l'équivalence syntaxique entre les segments, la possibilité d'échanger les segments de place, la possibilité de supprimer le MRP), nous avons proposé de nouvelles observations.

Plusieurs perspectives peuvent être proposées pour continuer ce travail. Tout d'abord, l'implication d'un autre annotateur et des séances de conciliation entre les annotateurs peuvent permettre d'obtenir des données de référence plus consensuelles. De même, d'autres MRP peuvent être considérés et une analyse comparative plus détaillée entre les emplois des MRP étudiés peut être effectuée. Pour la détection automatique de reformulations paraphrastiques, il est nécessaire d'améliorer la détection des répétitions et de tester une approche par apprentissage automatique pour le repérage de relations paraphrastiques et de segments en relation de paraphrase. D'autres indices encore peuvent être utilisés pour la détection de relations de paraphrase, en particulier ceux fournis par les informations paralinguistiques disponibles dans les transcriptions. Une autre perspective consiste à traiter les relations de paraphrase entre différents tours de parole, alors qu'actuellement nous le faisons au sein d'un même tour de parole uniquement. Nous voulons aussi comparer la reformulation paraphrastique telle qu'elle est effectuée à l'écrit et à l'oral : ce processus est d'une part similaire, car il consiste à éclaircir et faciliter la transmission et la compréhension de l'information, mais d'autre part, il est aussi différent quant à son processus cognitif (Hagège, 1985; Blanche-Benveniste *et al.*, 1991). Comme nous l'avons indiqué, les deux corpus exploités, tout en étant créés avec le même type de situation d'enregistrement d'entretiens semi-guidés, ont été constitués avec 40 ans de différence. Cela présente la possibilité de mener une analyse diachronique des MRP. Nous pouvons aussi étudier l'emploi des MRP en croisant les annotations avec les critères sociologiques des locuteurs. Finalement, nous envisageons de diffuser les données de référence constituées auprès des chercheurs.

Remerciements. Nous remercions Yoann Dupont pour son aide dans l'adaptation du logiciel SEM à nos corpus oraux et les relecteurs pour leur aide dans l'amélioration de la qualité du papier.

Références

ANDROUTSOPOULOS I. & MALAKASIOTIS P. (2010). A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, **38**, 135–187.

BANNARD C. & CALLISON-BURCH C. (2005). Paraphrasing with bilingual parallel corpora. In *ACL*, p. 597–604.

BARZILAY R. & MCKEOWN L. (2001). Extracting paraphrases from a parallel corpus. In *ACL*, p. 50–57.

BEECHING K. (2007). La co-variation des marqueurs discursifs bon, c'est-à-dire, enfin, hein, quand même, quoi et si vous voulez : une question d'identité ? *Langue française*, **154**(2), 78–93.

BHAGAT R. & HOVY E. (2013). What is a paraphrase ? *Computational Linguistics*, **39**(3), 463–472.

BLANCHE-BENVENISTE C., BILGER M., ROUGET C. & VAN DEN EYNDE K. (1991). *Le français parlé. Études grammaticales*. Paris : CNRS Éditions.

BOUAMOR H. (2012). *Étude de la paraphrase sous-phrastique en traitement automatique des langues*. Thèse de doctorat, Université Paris Sud, Paris.

BOUAMOR H., MAX A. & VILNAT A. (2012). Étude bilingue de l'acquisition et de la validation automatiques de paraphrases sous-phrastiques. *TAL*, **53**(1), 11–37.

BOUCHERON S. (2000). La langue de l'un, et celle de l'autre : l'entre parenthèses comme aire de reformulation. In *Répétition, Altération, Reformulation*, p. 113–118. Besançon : Presses Universitaires Franc-Comtoises.

CALLISON-BURCH C., COHN T. & LAPATA M. (2008). Parametric : An automatic evaluation metric for paraphrasing. In *COLING*, p. 97–104.

CHOMSKY N. (1975). *Reflections on language*. New-York, USA : Pantheon books.

COHEN J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, **20**(1), 37–46.

CULIOLI A. (1976). *Notes du séminaire de DEA, 1983-84*. Paris.

DAILLE B., HABERT B., JACQUEMIN C. & ROYAUTÉ J. (1996). Empirical observation of term variations and principles for their description. *Terminology*, **3**(2), 197–257.

DELÉGER L. & ZWEIGENBAUM P. (2008). Paraphrase acquisition from comparable medical corpora of specialized and lay texts. In *AMIA 2008*, p. 146–50.

ELHADAD N. & SUTARIA K. (2007). Mining a lexicon of technical terms and lay equivalents. In *BioNLP*, p. 49–56.

ESHKOL I., TELLIER I., DUPONT Y. & WANG I. (2014). Peut-on bien chunker avec de mauvaises étiquettes pos ? In *TALN 2014*.

ESHKOL-TARAVELLA I., BAUDE O., MAUREL D., HRIBA L., DUGUA C. & TELLIER I. (2012). Un grand corpus oral "disponible" : le corpus d'Orléans 1968-2012. *Traitement Automatique de Langues*, **52**(3), 17–46.

FLOTTUM K. (1995). *Dire et redire. La reformulation introduite par "c'est-à-dire"*. Thèse de doctorat, Hogskolen i Stavanger, Stavanger.

FRANÇOIS F. (1990). La communication inégale. Heurs et malheurs de l'interaction verbale. In *Actualités pédagogiques et psychologiques*. Neuchâtel-Paris : Delachaux & Niestlé.

FUCHS C. (1982). *La paraphrase*. Paris : PUF.

FUCHS C. (1994). *Paraphrase et énonciation*. Paris : Orphys.

FUJITA A. (2010). Typology of paraphrases and approaches to compute them. In *CBA to Paraphrasing & Nominalization*, Barcelona, Spain. Invited talk.

GRABAR N. & ZWEIGENBAUM P. (2000). A general method for sifting linguistic knowledge from structured terminologies. *JAMIASUP*, p. 310–314.

GULICH E. & KOTSCHI T. (1983). Les marqueus de la reformulation paraphrastique. *Cahiers de linguistique française*, **5**, 305–351.

GÜLICH E. & KOTSCHI T. (1987). Les actes de reformulation dans la consultation La dame de Caluire. In P. BANGE, Ed., *L'analyse des interactions verbales. La dame de Caluire : une consultation*, p. 15–81. Berne : P Lang.

HAGÈGE C. (1985). *L'homme de paroles. Contribution linguistique aux sciences humaines*. Paris : Fayard.

HAMON T., NAZARENKO A. & GROS C. (1998). A step towards the detection of semantic variants of terms in technical documents. In *International Conference on Computational Linguistics (COLING-ACL'98)*, p. 498–504, Université de Montréal, Montréal, Quebec, Canada.

HÖLKER K. (1988). *Zur Analyse von Markern*. Stuttgart : Franz Steiner.

HWANG Y. (1993). Eh bien, alors, enfin et disons en français parlé contemporain. *L'Information Grammaticale*, **57**, 46–48.

IBRAHIM A., KATZ B. & LIN J. (2003). Extracting structural paraphrases from aligned monolingual corpora. In *International Workshop on Paraphrasing*, p. 57–64.

KANAAN L. (2011). *Reformulations, contacts de langues et compétence de communication : analyse linguistique et interactionnelle dans des discussions entre jeunes Libanais francophones*. Thèse de doctorat, Université d'Orléans, Orléans.

KOK S. & BROCKETT C. (2010). Hitting the right paraphrases in good time. In *NAACL*, p. 145–153.

LANDIS J. & KOCH G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, **33**, 159–174.

LIN D. & PANTEL L. (2001). Dirt - discovery of inference rules from text. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, p. 323–328.

MADNANI N. & DORR B. J. (2010). Generating phrasal and sentential paraphrases : A survey of data-driven methods. *Computational Linguistics*, **36**, 341–387.

MADNANI N., RESNIK P., DORR B. & SCHWARTZ R. (2008). Applying automatically generated semantic knowledge : A case study in machine translation. In *NSF Symposium on Semantic Knowledge Discovery, Organization and Use*, p. 60–61.

MALAKASIOTIS P. & ANDROUTSOPOULOS I. (2007). Learning textual entailment using SVMs and string similarity measures. In *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, p. 42–47.

MARTIN R. (1976). *Inférence, antonymie et paraphrase*. Paris : Klincksieck.

MELČUK I. (1988). Paraphrase et lexique dans la théorie linguistique sens-texte in lexique et paraphrase. *Lexique*, **6**, 13–54.

MILICEVIC J. (2007). *La paraphrase : Modélisation de la paraphrase langagière*. Peter Lang.

OCH F. & NEY H. (2000). Improved statistical alignment models. In *ACL*, p. 440–447.

PASÇA M. & DIENES P. (2005). Aligning needles in a haystack : Paraphrase acquisition across the Web. In *IJCNLP*, p. 119–130.

PETIT M. (2009). *Discrimination prosodique et représentation du lexique : application aux emplois des connecteurs discursifs*. Thèse de doctorat, Université d'Orléans, Orléans.

QUIRK C., BROCKETT C. & DOLAN W. (2004). Monolingual machine translation for paraphrase generation. In *EMNLP*, p. 142–149.

ROSSARI C. (1990). Projet pour une typologie des opérations de reformulation. *Cahiers de linguistique française*, **11**, 345–359.

ROSSARI C. (1992). De l'exploitation de quelques connecteurs reformulatifs dans la gestion des articulations discursives. *Pratiques*, **75**, 111–124.

ROSSARI C. (1993). *Les opérations de reformulation. Analyse du processus et des marques dans une perspective contrastive français-italien*, In P. LANG, Ed., *Sciences pour la communication*.

ROULET E. (1987). Complétude interactive et connecteurs reformulatifs. *Cahiers de linguistique française*, **8**, 111–140.

SAUNIER E. (2012). Disons : un impératif de dire ? Remarques sur les propriétés du marqueur et son comportement dans les reformulations. *L'Information Grammaticale*, **132**, 25–34.

SEKINE S. (2005). Automatic paraphrase discovery based on context and keywords between NE pairs. In *International Workshop on Paraphrasing*, p. 80–87.

SHEN S., RADEV D., PATEL A. & ERKAN G. (2006). Adding syntax to dynamic programming for aligning comparable texts for the generation of paraphrases. In *ACL-COLING*, p. 747–754.

SHINYAMA Y., SEKINE S., SUDO K. & GRISHMAN R. (2002). Automatic paraphrase acquisition from news articles. In *Proceedings of HLT*, p. 313–318.

TESTON-BONNARD S. (2008). Je veux dire est-il toujours une marque de reformulation ? In M. L. BOT, M. SCHUWER & E. RICHARD, Eds., *Rivages linguistiques. La Reformulation. Marqueurs linguistiques. Stratégies énonciatives*, p. 51–69. Rennes : PUR.

VEZIN L. (1976). Les paraphrases : étude sémantique, leur rôle dans l'apprentissage. *L'année psychologique*, **76**(1), 177–197.

VILA M., ANTÒNIA MART M. & RODRÍGUEZ H. (2011). Paraphrase concept and typology. a linguistically based and computationally oriented approach. *Procesamiento del Lenguaje Natural*, **46**, 83–90.

Evaluation d'une approche de classification possibiliste pour la désambiguïsation des textes arabes

Raja Ayed[1] Ibrahim Bounhas[2] Bilel Elayeb[1, 3] Narjès Bellamine Ben Saoud[1, 4] Fabrice Evrard[5]

(1) Laboratoire de recherche RIADI, ENSI, Université de la Manouba, 2010, Tunisie
(2) Laboratoire de l'informatique pour les systèmes industriels, Institut Supérieur de Documentation, Université de la Manouba, 2010,Tunisie
(3) Institut de technologies des Émirats, P.O. Box: 41009, Abu Dhabi, Émirats arabes unis
(4) Institut supérieur de l'informatique, ISI, Université de Tunis El Manar, 1002, Tunisie
(5) Institut de recherche en informatique de Toulouse (IRIT), 02 rue Camichel, 31071 Toulouse, France

ayed.raja@gmail.com, bounhas.ibrahim@yahoo.fr, bilel.elayeb@riadi.rnu.tn, narjes.bellamine@ensi.rnu.tn, fabrice.evrard@enseeiht.fr

Résumé. La désambiguïsation morphologique d'un mot arabe consiste à identifier l'analyse morphologique appropriée correspondante à ce mot. Dans cet article, nous présentons trois modèles de désambiguïsation morphologique de textes arabes non voyellés basés sur la classification possibiliste. Cette approche traite les données imprécises dans les phases d'apprentissage et de test, étant donné que notre modèle apprend à partir de données non étiquetés. Nous testons notre approche sur deux corpus, à savoir le corpus du Hadith et le Treebank Arabe. Ces corpus contiennent des données de types différents classiques et modernes. Nous comparons nos modèles avec des classifieurs probabilistes et statistiques. Pour ce faire, nous transformons la structure des ensembles d'apprentissage et de test pour remédier au problème d'imperfection des données.

Abstract. Morphological disambiguation of Arabic words consists in identifying their appropriate morphological analysis. In this paper, we present three models of morphological disambiguation of non-vocalized Arabic texts based on possibilistic classification. This approach deals with imprecise training and testing datasets, as we learn from untagged texts. We experiment our approach on two corpora i.e. the Hadith corpus and the Arabic Treebank. These corpora contain data of different types: traditional and modern. We compare our models to probabilistic and statistical classifiers. To do this, we transform the structure of the training and the test sets to deal with imprecise data.

Mots-clés : Traitement Automatique des Langues Naturelles, Désambiguïsation Morphologique de l'Arabe, Théorie des Possibilités, Classification Possibiliste.

Keywords: Natural Language Processing, Arabic Morphological Disambiguation, Possibility Theory, Possibilistic Classification.

1 Introduction

De nombreux mots Arabes possèdent la même forme orthographique. Ceci est dû à la richesse morphologique de cette langue (Diab et al., 2004). En effet, l'omission des voyelles courtes peut générer plus de 12 interprétations morphologiques d'un mot donné (Habash et Rambow, 2007). Par conséquent, l'une des formes d'ambiguïté les plus relevées en arabe est l'ambiguïté morphologique. Un mot peut être ambigu à l'égard de sa structure interne. Le traitement morphologique porte sur le morphème qui constitue l'unité élémentaire discernable. L'analyse morphologique d'un mot a pour rôle de déterminer les valeurs d'un grand nombre de caractéristiques ou d'attributs morphologiques d'une entité lexicale (un mot), comme la catégorie grammaticale (nom, verbe, etc.), le genre, le nombre, etc. En fait, un mot non voyellé peut conduire à de nombreuses solutions morphologiques. Par exemple, le mot وقف (wqf), en dehors du contexte, peut être interprété comme وَقَفَ (waqafa, "il s'est levé") ou وَقْفٌ (waqfun, "cession") ou

encore وَقِفْ (waqif, "et lève-toi"), où ce mot est une concaténation de la conjonction و "et" avec le verbe قِفْ "se lever" qui est conjugué à l'impératif. Malgré leur importance, les voyelles courtes ne sont utilisées que dans les textes religieux (Coran, Hadith, etc.) et les manuels didactiques contrairement aux textes modernes trouvés dans les journaux et dans les livres.

L'ambiguïté morphologique se manifeste lorsque l'analyse associe, à une unité lexicale, plusieurs informations non-conformes au contexte du mot, autrement dit quand l'analyse fournit plusieurs valeurs pour certains attributs morphologiques (Hajic, 2000). Par ailleurs, une approche pour la désambiguïsation morphologique arabe est nécessaire pour faire face à l'ambiguïté des mots non voyellés. La désambiguïsation consiste, donc, à attribuer la valeur exacte d'un attribut morphologique parmi celles proposées par l'analyseur. De nombreux travaux utilisent des approches de classification pour résoudre la tâche morphologique de désambiguïsation (Roth et al., 2008).

Nous discutons dans ce papier la contribution d'une nouvelle approche pour la désambiguïsation morphologique arabe basée sur la classification possibiliste. Le but principal est d'apprendre des dépendances morphologiques à partir des textes voyellés et de tester sur des textes non voyellés. Nous organisons ce document comme suit. Tout d'abord, dans la section 2, nous présentons brièvement un état de l'art sur la désambiguïsation morphologique arabe. Quant à la section 3, elle est consacrée à un résumé sur la théorie des possibilités. Notre approche pour la désambiguïsation morphologique possibiliste est détaillée dans la section 4. Les résultats expérimentaux sont présentés et discutés dans la section 5. Nous concluons, dans la section 6 et nous proposons quelques pistes pour de futures recherches.

2 La désambiguïsation morphologique arabe

Plusieurs travaux conduisent la désambiguïsation des mots arabes, d'un texte, à l'identification de leurs catégories grammaticales (POS- *part-of-speech*). La désambiguïsation de POS est le fait de déterminer la catégorie grammaticale d'un mot par son utilisation dans un contexte particulier. Elle peut, également, être considérée comme un problème de classification: l'ensemble des valeurs de POS présentent les classes et une méthode de classification est utilisée pour attribuer à chaque occurrence d'un mot (analyse d'un mot) une classe sur la base de la certitude du contexte. L'une des étapes importantes dans la désambiguïsation est la sélection de la méthode de classification. Des méthodes de classification automatique supervisée ont été appliquées. Elles utilisent des techniques d'apprentissage pour apprendre un classifieur à partir des ensembles d'apprentissage annotés (les valeurs de la classe POS sont identifiées). Dans la littérature, les approches de désambiguïsation, se répartissent en trois catégories. Principalement, ces approches sont: les approches à base de règles, les approches statistiques et les approches hybrides qui combinent les deux dernières.

2.1 Les approches à base de règles

Les approches à base de règles sont, encore, dites linguistiques. Elles utilisent une base de connaissances des règles écrites par des linguistes permettant d'attribuer des étiquettes aux différentes catégories morphologiques (Daoud, 2009 ; Othman et al., 2004). Nous parlons, principalement, des heuristiques, des règles contextuelles et des règles non contextuelles (Elshafei et al., 2002). Les arbres de décision (Quinlan, 1986) sont conçus pour exposer des bases de règles. Un arbre de décision est un modèle prédictif utilisé pour représenter les règles de classification avec une structure en arbre qui partitionne de façon récursive l'ensemble de données d'apprentissage. Chaque nœud interne d'un arbre de décision représente un test sur une valeur d'un attribut de classification, et chaque branche représente un résultat de test. Une prédiction est faite quand un nœud feuille est atteint. Cette approche est étendue pour extraire et calculer des mesures statistiques utilisées pour l'étiquetage grammatical (Schmid et al., 1994).

2.2 Les approches statistiques

Les approches statistiques forment des modèles d'apprentissage à partir des corpus annotés. Elles incorporent des méthodes de classification telles que les modèles de Markov cachés (Garside et Leech, 1987), SVM (Vapnik, 1998), etc. pour calculer des taux de probabilité de chaque valeur résultante d'une catégorie grammaticale d'un mot. Un modèle peut être utilisé pour classer automatiquement les autres textes en se référant aux taux déjà calculés. (Diab et al., 2004) développent un classifieur morphologique utilisant SVM. Ils entrainent et testent le classifieur sur un Treebank arabe de 4000 phrases d'apprentissage et 100 phrases de test. (Habash et Rambow, 2005) utilisent SVM en se basant sur des informations fournies à partir d'un analyseur morphologique. (Mansour et al., 2007) combinent les probabilités calculées sur des ensembles d'apprentissage Arabes et Hébreux pour classer les catégories grammaticales des mots des textes arabes. Ils utilisant les mêmes paramètres de test de (Diab et al., 2004). Quelques travaux de recherches comprennent les modèles de Markov cachés (HMM). (ElHadj et al., 2009) présentent un système d'étiquetage grammaticale qui combine l'analyse morphologique et le modèle de Markov. L'étiqueteur se base sur la structure de la phrase arabe. Dans un premier lieu, le texte est entièrement analysé morphologiquement pour réduire le nombre de valeurs possibles de POS. Dans un second lieu, le modèle statistique (HMM), fondé sur la structure de la phrase arabe,

est utilisé pour attribuer à chaque mot la valeur exacte de sa catégorie grammaticale. (ElHadj et al., 2009) ont utilisé leur propre corpus annoté qui est composé de vieux livres arabes. Le total des mots, dans ce corpus, est environ 21000 mots.

2.3 Les approches hybrides

Une approche hybride combine les règles linguistiques avec les informations statistiques afin de résoudre l'ambiguïté morphologique. Dans (Tlili-Guiassa, 2006), on propose une approche qui analyse les affixes grammaticaux et flexionnels et les règles grammaticales en se basant sur l'approche MBL (*Memory based learning*) (Lin et al., 1994). Elle est appliquée pour classer une collection de textes coraniques et éducatifs. (Zribi et al., 2006) combinent l'approche à base de règles avec un étiqueteur trigramme HMM (Collins, 2002). L'apprentissage du classifieur trigramme a été fait sur des textes comportant 6000 mots. Des règles heuristiques ont été appliquées pour sélectionner parmi les résultats proposés.

(Khoja, 2001) a mis en œuvre une approche hybride qui utilise l'algorithme de Viterbi (Forney, 1973; Fettweis et Meyr, 1991). Elle calcule deux probabilités sur un corpus annoté composé de 50000 mots: (i) une probabilité lexicale, qui est la probabilité qu'un mot ait une certaine valeur d'un attribut morphologique spécifié, et (ii) une probabilité contextuelle, qui est la probabilité d'une étiquette à suivre une autre. Une liste de règles grammaticales est préparée à partir de ces statistiques dans le but d'assurer plus de 90% de précision.

Les outils de désambiguïsation linguistiques sont plus rapides et plus efficaces et fiables que les outils statistiques (Hoceini et al., 2011). L'approche linguistique qui n'a besoin que de l'intervention manuelle d'un linguiste, définit un ensemble de règles spécifiques à un domaine particulier. Alors que, les statistiques calculées pour l'apprentissage sont appliquées à n'importe quel domaine de test. Néanmoins, les deux approches statistiques et hybrides nécessitent une phase d'apprentissage dans le but est d'apprendre les paramètres requis pour la désambiguïsation. Par conséquent, l'approche hybride est considérée comme la plus efficace et cohérente en termes d'analyse, car elle combine les deux approches et tire profit de leurs avantages.

La plupart des désambiguïseurs morphologiques arabes ne traitent que la catégorie grammaticale (POS). Les travaux récents (Habash et al., 2009 ; Ayed et al., 2012b) définissent 14 attributs qui décrivent les caractéristiques morphologiques d'un mot. Nous étendons, dans cet article, la classification à ces 14 attributs morphologiques.

3 La théorie des possibilités

La théorie des possibilités a été introduite par Zadeh en 1978 pour palier au problème de l'imperfection des données et de l'incomplétude de l'information (Dubois, Prade, 1994). Une information est imparfaite lorsqu'elle est incertaine et/ou imprécise. Nous décrivons, dans les paragraphes suivants, les fonctions, les mesures et les degrés utilisés pour traduire l'incertitude et l'imprécision des données dans la théorie des possibilités.

3.1 La distribution de possibilité

La théorie des possibilités est fondée sur la notion de distribution des possibilités désignée par π. Cette distribution correspond à une application de l'univers de discours $\Omega=\{\omega_1, \omega_2,\ldots, \omega_n\}$ vers l'intervalle [0, 1] modélisant les connaissances du monde réel. Elle distingue les états (les ω_i) plausibles et les états peu plausibles. Les valeurs de cette application sont appelées degrés de possibilités. Si un degré est égal à 1, alors l'état ω_i associé est plausible. Toutefois, si ce degré est égal à 0 alors l'état est dit impossible.

3.2 Les mesures de possibilité et de nécessité

L'imprécision se manifeste quand un état de la réalité est décrit par une variable propositionnelle de valeurs multiples. L'incertitude traduit le fait de ne pas connaître ou prévoir un état de la réalité pour déterminer la valeur de vérité d'une proposition (Dubois et Prade, 1994). Nous évaluons un état par le calcul de deux mesures qui sont, respectivement, la possibilité et la nécessité. Nous désignons *A* un sous-ensemble d'états de l'univers du discours Ω. Nous décrivons la mesure de possibilité de *A*, moyennant une distribution de possibilités π (définie sur Ω), comme suit:

$$\Pi(A) = \max_{\omega \in A} \pi(\omega) \quad (1)$$

La mesure de nécessité est extraite à partir de la mesure de possibilité et elle est décrite par:

$$N(A) = \min_{\omega \notin A}[1 - \pi(\omega)] = 1 - \Pi(\bar{A}) \quad (2)$$

Dans la formule 2, $\bar{A}$ définit le complément de A en d'autres termes il englobe les éléments de Ω qui n'appartiennent pas à A. Π(A) évalue le degré de consistance de l'événement *A*. N (A) estime dans quelle mesure *A* est certainement déduit par la connaissance représentée par π. La mesure de nécessité définit le degré auquel on attend l'occurrence d'un événement (Dubois et Prade, 1985).

4 L'approche possibiliste de désambiguïsation morphologique

Nous proposons une approche de désambiguïsation morphologique, des textes arabes, basée sur la théorie des possibilités. Plusieurs travaux utilisent les approches de classification pour résoudre l'ambiguïté morphologique (Habash et Rambow, 2005). Un mot est considéré ambigu si l'analyseur morphologique fournit plus qu'une seule solution pour ses attributs morphologiques. La classification assigne une classe à une instance de test donnée. La tâche de désambiguïsation consiste, donc, à accorder à un mot ambigu les valeurs des attributs morphologiques appropriées. Elle est divisée en deux grandes phases qui sont l'apprentissage et le test. Les résultats d'analyse morphologique donnés par les mots voyellés sont, généralement, moins ambigus que ceux donnés par les mots non voyellés. Ainsi, nous proposons d'apprendre à partir des textes voyellés et de tester sur des textes non voyellés.

Pour ce faire, nous commençons par définir l'ensemble d'apprentissage. Cet ensemble est constitué d'une liste d'instances qui sont caractérisées par des attributs avec des valeurs de classes connues. Par conséquent, pour résoudre l'ambiguïté de la catégorie grammaticale (par exemple), nous déterminons d'abord les attributs appropriés qui décrivent chaque instance. En nous inspirant de la technique de classification Yamcha (Diab et al., 2004), nous estimons qu'un attribut morphologique d'un mot est fortement lié à celui des mots qui le précèdent et le suivent. Nous définissons une fenêtre qui contrôle le nombre de mots (avant et après) considérés comme des attributs décrivant la classe d'une instance. Dans des approches existantes, la taille de la fenêtre est 2 (Habash et Rambow, 2005). Notre modèle applique une fenêtre avec une taille quelconque. Pour classer la catégorie grammaticale (POS- *part-of-speech*) d'un mot particulier, si la fenêtre est de 2, nous définissons les attributs POS-2, POS-1, POS+1 et POS+2. Ils indiquent, respectivement, les catégories grammaticales des deux mots précédents et des deux mots suivants. POS peut être décrit par l'ensemble des autres attributs morphologiques, en plus du POS. Nous pouvons utiliser, par exemple, les attributs genre-2, genre-1, nombre+1, nombre+2, etc. La valeur de la classe est la catégorie grammaticale du mot courant. A cet effet, nous identifions, pour chaque mot d'un texte voyellé, 14 attributs morphologiques qui sont *POS, conjonction, particule, déterminant, pronom, personne, voix, aspect, genre, nombre, cas, préposition, mode* et *adjectif*. Ces attributs sont calculés par l'analyseur morphologique Aramorph (Ayed et al., 2012b). Ayant l'exemple de la phrase suivante الرَازِي وَالْبَغْدَادِي دَرَسَا عُلُومَ الطِّبِ (Al-Razi et Al-Bagdadi ont étudié les sciences de la médecine), nous déterminons, l'instance du tableau 1, associée au mot «دَرَسَا (ont étudié)». Pour cette instance, la classe est la catégorie grammaticale (POS) et les attributs utilisés sont les catégories grammaticales des 2 mots adjacents.

POS-2	**POS-1**	**POS+1**	**POS+2**	**POS**
NOM_PROPRE الرَازِي (Al-Razi)	NOM_PROPRE وَالْبَغْدَادِي (et Al-Bagdadi)	NOM عُلُومَ (les sciences)	NOM الطِّبِ (de la médecine)	VERBE دَرَسَا (ont étudié)

TABLEAU 1 : L'instance reliée au mot «دَرَسَا»

L'analyse morphologique d'un mot est fournie indépendamment de son contexte. Dans un texte arabe, même les mots voyellés peuvent donner une analyse morphologique ambigüe. La forme voyellée «إِبْنِ » fournit des valeurs de l'attribut POS à savoir un verbe (tu construis) et un nom (fils de). Par conséquent, les instances d'apprentissage peuvent fournir des informations incomplètes. Ces informations sont dites imprécises lorsque les attributs et ou la classe donnent plus qu'une seule valeur.

Nous pouvons affirmer, clairement, que le contexte nécessaire pour lever l'ambiguïté d'un mot donné est lui-même ambigu ce qui est considérée comme un cas d'imprécision. En effet, la théorie des probabilités est incapable de traiter un tel type de données (imprécises), alors que la théorie des possibilités s'applique naturellement à ces problèmes. Nous proposons des modèles d'apprentissage et de test (classification) basés sur la théorie des possibilités.

4.1 L'apprentissage possibiliste des attributs morphologiques

Dans la phase d'apprentissage, nous formons un classificateur pour chaque attribut morphologique. Autrement, nous instaurons un ensemble d'apprentissage pour chaque attribut morphologique. Nous obtenons, globalement, 14 ensembles. Chacun est décrit par les attributs AM ± i où AM forme la totalité des attributs morphologiques et *i* constitue la taille de la fenêtre. Si cette taille est égale à 2, nous obtenons 56 (14x4) attributs d'apprentissage. A chaque mot voyellé est liée une instance décrite par les valeurs de ces 56 attributs et dont la classe est reconnue. Cette classe est l'attribut morphologique associé à l'ensemble d'apprentissage.

Nous devons prendre en compte le fait que les attributs et/ou les classes des instances de classification sont imprécises autrement dit ayant plus qu'une seule valeur possible. L'imprécision est gérée par des distributions de possibilités désignées par π. Soit T un ensemble de données d'apprentissage et I_k l'ensemble des valeurs des attributs de l'instance k. On note également A_j le $j^{ième}$ attribut de cet ensemble et a_{jL} une valeur possible de A_j. Nous nous inspirons des travaux de Haouari et al. (Haouari et al., 2009) et le modèle de recherche d'information possibiliste développé par Bounhas et al. (2011) pour calculer la fréquence normalisée d'une valeur d'un attribut a_{jL} pour une classe c_i comme suit:

$$Freq\ (a_{jL}, c_i) = \frac{Occ(a_{jL}, c_i)}{Max_{l=1}^{|A_j|} Occ(a_{jl}, c_i)} \quad (3)$$

$Occ(a_{jL}, c_i)$ indique le nombre d'occurrences de la classe c_i avec la valeur a_{jL} c.à.d. le nombre d'instances dont la classe est égale à c_i et la valeur a_{jL} est une valeur possible de l'attribut A_j. $|A_j|$ est le nombre de valeurs possibles de A_j. Nous utilisons l'opérateur MAX pour obtenir les fréquences normalisées (Bounhas et al., 2011). La somme de toutes les fréquences associées à une classe c_i n'est pas égale à 1 ce qui est l'une des principales hypothèses de la théorie des possibilités afin de traiter des données imparfaites. Dans le cas de l'imperfection des données, le nombre d'occurrences d'une valeur d'un attribut est flou. Nous introduisons une mesure β_{jk} appelée le taux de l'imprécision de l'attribut A_j dans l'instance I_k (Haouari et al., 2009). Le nombre d'occurrences est calculé suivant la formule 4 :

$$Occ(a_{jL}, c_i) = \sum_{k=1}^{|T|} \beta_{jk} * \emptyset_{ijkL} \quad (4)$$

Le taux $\beta_{jk} = 1/N$ où N est le produit de $|A_{jk}|$ et $|C_k|$. Ces derniers représentent, respectivement, le nombre de valeurs de A_j dans l'instance I_k et le nombre de classes possibles de I_k. Si l'instance est parfaite, alors β_{jk} =1. Si dans une instance donnée, un attribut possède deux valeurs et la classe a une seule valeur alors le taux de l'imprécision est égal à 0.5. $\emptyset_{ijkL}$ est égale à 1 si la valeur a_{jL} appartient aux valeurs possibles de A_j dans l'instance I_k, et la classe c_i appartient aux valeurs de classes de I_k et 0 sinon.

Les fréquences normalisées sont calculées pour la totalité des instances des différents ensembles d'apprentissage. Elles traduisent les distributions de possibilités de chaque attribut par rapport à une classe.

4.2 La classification possibiliste des attributs morphologiques

La classification des 14 attributs morphologiques consiste à désambiguïser chaque mot non voyellé en lui associant les valeurs correctes et précises de ces attributs. Pour ce faire, nous commençons par préparer les instances de l'ensemble de test. En effet, chaque instance décrit un mot non voyellé d'un texte par des attributs de classification qui représentent les mêmes attributs d'apprentissage c.à.d. $AM \pm i$. La classe de l'instance est la valeur correcte à identifier de l'attribut morphologique. Le tableau 2 décrit une instance de test dont l'attribut morphologique à classer est le POS. Pour simplifier la représentation de l'instance, nous nous contentons de 4 attributs de classification à savoir DET-2, POS-1, CONJONCTION-1 et POS+2. Elle est réellement décrite par les 56 attributs. Cette instance est imprécise puisqu'elle donne deux valeurs possibles de l'attribut POS-1.

DETERMINANT-2	POS-1	CONJONCTION-1	POS+2	…	POS
DET	{VERBE; NOM_PROPRE}	NCONJ	NOM	…	?

TABLEAU 2 : Un exemple d'une instance de test imprécise

Nous calculons la possibilité de chaque classe c_i par rapport à une instance imparfaite I_k ayant m attributs. Cette mesure s'inspire du classifieur possibiliste de Haouari et al., (2009). La mesure de possibilité est le produit des fréquences de tous les attributs calculées par rapport à l'ensemble d'apprentissage. Cependant, un facteur spécifique est ajouté pour les attributs imprécis. Ce facteur est le taux de l'imprécision β_{jk}. Par exemple, si un attribut a quatre valeurs possibles, nous calculons le produit des fréquences de ces quatre valeurs et nous introduisons le taux β_{jk} égal à ¼. Ainsi, la mesure de possibilité est donnée par la formule 5 :

$$\Pi(c_i|I_k) = \prod_{j=1}^{m} \prod_{L=1}^{|A_{jk}|} Freq\ (a_{jL}, c_i) * \beta_{jk} \quad (5)$$

En se référant à l'instance du tableau 2, si la classe POS possède trois valeurs possibles i.e. NOM, VERBE et NOM_PROPRE, alors trois mesures de possibilités sont à calculer par rapport à cette instance. Ces mesures sont $\Pi(POS = NOM|I_k)$, $\Pi(POS = VERBE|I_k)$ et $\Pi(POS = NOM\_PROPRE|I_k)$. Pour déterminer la mesure $\Pi(POS = NOM|I_k)$, les fréquences nécessaires sont Freq(DETERMINANT-2=DET, POS=NOM), Freq(POS-1=VERBE, POS=NOM), Freq(POS-1=NOM_PROPRE, POS=NOM), etc. Ces fréquences sont calculées dans la phase d'apprentissage.

Un classifieur possibiliste a été défini dans (Ayed et al., 2012a) qui n'évalue pas le pouvoir discriminant des valeurs d'un attribut, car il utilise uniquement la mesure de possibilité (formule 5). Cependant, nous pouvons découvrir que certaines valeurs, d'un attribut donné, ont un plus grand impact dans la résolution de la bonne classe. La théorie des possibilités modélise cet effet par la mesure de nécessité. Elle détermine le degré auquel on attend l'occurrence d'un événement (Elayeb et al, 2009). Cette mesure est donnée par la formule suivante :

$$N(c_i|I_k) = 1 - \prod_{j=1}^{m} \prod_{L=1}^{|A_{jk}|} \left(1 - \frac{\lambda ijL\ * Freq\ (a_{jL}, c_i)}{\beta_{jk}}\right) \quad (6)$$

Où $\boldsymbol{\lambda}_{ijL} = \log_{10}(P/nC_{jL})$

Avec P est le nombre de classes possibles et nC_{jL} est le nombre de classes ayant une fréquence non nulle avec la valeur de la valeur a_{jL} ou en d'autres termes $\text{Freq}\ (a_{jL}, c_i) > 0$.

Nous définissons trois modèles de classification pour déterminer la valeur appropriée d'un attribut morphologique. Le premier modèle se base uniquement sur le calcul des mesures de possibilités. Le deuxième modèle se base sur les mesures de nécessité. Le troisième étant une combinaison des deux, il utilise la somme des mesures de possibilité et de nécessité. La classe choisie correspond à la valeur c*. La meilleure classe de l'instance I_k est celle ayant le plus grand score parmi toutes les classes:

$$c^* = \arg\max_{c_i}(\ \text{score}\ (c_i|I_k) * score(c_i|w_k)) \quad (23)$$

Dans cette formule, score (c_i | I_k) peut être égal à $\Pi(c_i|I_k)$ ou $N(c_i|I_k)$ ou $\Pi(c_i|I_k) + N(c_i|I_k)$. Nous introduisons la score lexical score ($c_i|w_k$) (Jurafsky, Martin, 2009). Cette mesure calcule le degré de dépendance d'un mot w_i avec une classe particulière c_i dans l'ensemble d'apprentissage. Si w_i est le mot de l'instance de test I_k, alors la possibilité lexicale répond à la question : si nous nous attendions que c_i soit la classe de I_k, quelle est la possibilité que le mot soit w_i ? De même ce score peut être calculé de trois manières différentes en utilisant la possibilité et/ou la nécessité.

4.3 La classification non possibiliste des attributs morphologiques

Nous visons à comparer les résultats de la classification possibiliste avec les résultats donnés par des classifieurs non possibilistes, afin de désambiguïser les attributs morphologiques. Ces classifieurs ne traitent pas les données imparfaites. Par conséquent, nous proposons de transformer la structuration des données des ensembles d'apprentissage et de test, afin de les préparer pour qu'elles soient utilisées par des classifieurs non possibilistes. Les nouveaux attributs doivent donner des informations précises. Pour ce faire, nous commençons par présenter un ensemble de données imparfaites. La figure 1(a) donne un exemple d'un ensemble d'apprentissage. Nous supposons que la classe à désambiguïser est POS et que les attributs utilisés, pour l'apprentissage et le test, sont POS-1 et CONJUNCTION+1. Cet ensemble est composé de deux instances. La première instance est imprécise, car elle fournit deux valeurs possibles de la classe (NOM_PROPRE et VERBE). La deuxième instance est imprécise puisqu'elle fournit deux valeurs de

l'attribut POS-1 (NOM, VERBE). Nous transformons la structure de données afin d'obtenir un ensemble parfait sans perdre les informations qui s'y trouvent. Pour résoudre le problème de l'imprécision, nous désignons les valeurs, de l'attribut *A*, par $A_i = \{a_1, a_2, ..., a_n\}$. Nous constituons de nouveaux attributs. En effet, nous associons l'attribut *A* à chacune de ses valeurs a_i pour former des attributs notés "*A_a_i*". Ainsi, l'attribut POS-1 a deux valeurs possibles (NOM et VERBE) dans l'ensemble de données de la figure 1. Nous obtenons donc deux attributs POS- 1_NOM et POS-1_VERBE. Nous accordons, aux nouveaux attributs, des valeurs binaires (oui ou non). Pour une instance donnée, si a_i appartient à une des valeurs de l'attribut, *A* alors l'attribut "A_a_i" est égal à "oui". A partir des données de la figure 1 (a), nous formons un nouvel ensemble de données précises (voir figure 1 (b)).

Pour résoudre l'imprécision des classes, nous décomposons une instance en plusieurs ayant chacune une seule valeur de la classe. Si une instance possède *n* valeurs possibles de la classe $\{c_1, c_2, \ldots, c_n\}$, alors nous obtenons *n* instances dont les valeurs des attributs sont similaires. Nous associons à chaque instance une valeur c_i.

Les instances dont la classe est précise (ayant une seule valeur) seront dupliquées afin d'augmenter leur poids dans le calcul des mesures de classification. La figure 1(c) présente un ensemble de données parfaites générées à partir des instances de la figure 1(a). Pour lever l'ambiguïté des textes non voyellés moyennant les approches non possibilistes, nous utilisons les méthodes SVM (Vapnik, 1998), le modèle bayésien naïf (Pearl, 1988) et les arbres de décision (Quinlan, 1986). Nous alignons les données au format d'entrée du logiciel WEKA[1]. Ce logiciel fournit des algorithmes d'apprentissage automatique et donne leurs résultats de classification. Nous utilisons WEKA pour classer les attributs morphologiques selon les modèles SVM, les arbres de décision et le modèle bayésien naïf.

POS-1	CONJONCTION+1	POS
NOM	NCONJ	{NOM_PROPRE; VERBE}
{NOM; VERBE}	CONJ	NOM

(a) Instances Imparfaites

POS-1_NOM	POS-1_VERBE	CONJONCTION+1_ CONJ	CONJONCTION+1_ NCONJ	POS
Oui	Non	Non	Oui	{NOM_PROPRE; VERBE}
Oui	Oui	Oui	Non	NOM

(b) Instances dont les attributs sont précis et les classes sont incertaines

POS-1_NOM	POS-1_VERBE	CONJONCTION+1_CONJ	CONJONCTION+1_NCONJ	POS
Oui	Non	Non	Oui	NOM_PROPRE
Oui	Non	Non	Oui	VERBE
Oui	Oui	Oui	Non	NOM
Oui	Oui	Oui	Non	NOM

(c) Instances parfaites

FIGURE 1 : Transformation des instances imparfaites en des instances parfaites

5 Expérimentations

Dans ce paragraphe, nous décrivons les corpus utilisés pour nos expérimentations. Nous présentons la méthode d'évaluation et les résultats expérimentaux mettant en évidence les aspects de classification possibiliste et non possibiliste.

5.1 Les collections de test

L'objectif principal de notre approche est d'acquérir des dépendances morphologiques à partir des textes voyellés et de tester sur des textes non voyellés. En outre, nous considérons les textes arabes classiques, qui ont été ignorés dans des travaux connexes précédents. Par conséquent, nous utilisons une collection d'histoires arabes "Hadiths" qui a fait le

[1] http://weka.wikispaces.com/

sujet de plusieurs travaux (Bounhas et al., 2011 ; Harrag et al., 2013), etc. Les Hadiths parlent de toutes les préoccupations du monde réel et couvrent des connaissances communes et universelles. Pour justifier notre choix, nous estimons que le corpus de hadiths est l'un des rares corpus arabes voyellés. Il contient environ 1400 livres voyellés de hadith, chacun comporte des milliers d'histoires arabes. Les six livres les plus reconnus comprennent plus de 2,5 millions de mots et plus de 95 000 fragments (titres et paragraphes). Par ailleurs, ce corpus est bien structuré et les titres des chapitres et des sous-chapitres représentent des informations contextuelles pertinentes pour désambiguïser des textes (Bounhas et al., 2011). Parmi les textes du corpus de hadiths, nous utilisons six livres encyclopédiques, regroupés par thèmes, qui sont Sahih Al-Bukhari, Sunan Abi Dawud, Sunan Ettermidhi, Sunan Ibn Majah, Sunan Annasaii et Sahih Muslim (Ayed et al., 2012a).

Nous menons nos expérimentations également sur le corpus Arabic Treebank (ATB part 2 v2.0) (Maamouri et al., 2009). Il s'agit d'un corpus de textes arabes non voyellés qui a été produit par *Linguistic Data Consortium*. Ce corpus comprend plus de 500 articles du journal égyptien Al Oumma. Il contient environ 144K de mots annotés (un mot est annoté si on indique la valeur de sa catégorie grammaticale).

Les corpus utilisés présentent deux types de textes i.e. modernes et classiques. Pour pouvoir apprendre les dépendances morphologiques du Hadith, nous passons par l'analyseur morphologique des textes voyellés Aramorph. Cet analyseur nous fournit les valeurs des 14 attributs morphologiques. L'annotation du corpus Arabic Treebank nous donne les valeurs de l'attribut POS. Le test (ou la classification) se fait directement sur les textes non voyellés de Arabic Treebank. Quant aux textes de Hadith, une étape d'élimination des voyelles courtes est indispensable pour pouvoir tester sur des textes non voyellés.

Pour évaluer les résultats des classifications possibilistes et non possibilistes, nous utilisons la méthode de la validation croisée (Kohavi, 1995). En effet, nous formons 10 itérations pour chaque texte du corpus: 90% d'un texte voyellé est utilisé pour l'apprentissage et 10% de mots de ce texte seront classés après avoir éliminé leurs voyelles courtes.

5.2 Les résultats expérimentaux

Pour classer les 14 attributs morphologiques, nous procédons comme suit : Tout d'abord nous analysons les textes voyellés de Hadith et nous sauvegardons les solutions morphologiques de chaque attribut. Nous formons, pour tout attribut morphologique A, un ensemble d'apprentissage. A chaque mot voyellé est associée une instance. Les instances de cet ensemble sont décrites par les attributs $AM \pm i$ (voir section 4.1) et la classe est l'attribut morphologique A. Nous aurons 14 ensembles d'apprentissage. Nous supprimons, par la suite, les voyelles courtes des mêmes textes. Nous formons de la même manière des ensembles de test décrites par les mêmes attributs que les ensembles d'apprentissage. Les valeurs de classes de leurs instances sont non reconnues (ambigües). Elles constituent les attributs morphologiques à classer. Nous désambiguïsons, ensuite, chaque mot de ces textes avec nos trois modèles de classification possibiliste. Pour ce faire, nous calculons les mesures de possibilité et de nécessité en se référant aux fréquences calculées par rapport aux ensembles d'apprentissage (voir section 4). Nous comparons les résultats obtenus avec ceux donnés par les mots voyellés. Pour classer les 14 attributs morphologiques en utilisant les classifieurs non possibilistes, nous utilisons les mêmes structures des instances d'apprentissage et de test.

Les approches non-possibilistes ne supportent pas l'imperfection des données. Nous les transformons en des données parfaites (voir section 4.3) et nous les adaptons au format d'entrée de l'outil Weka pour qu'elles soient appliquées sur des algorithmes de classification de SVM, Arbres de décision et les classifieurs Bayésiens Naïfs. Le tableau 3 présente les taux de désambiguïsation des 14 attributs morphologiques.

Les expérimentations prouvent que les classifieurs possibilistes donnent de meilleurs taux de désambiguïsation par rapport aux classifieurs SVM, Bayésien Naïf et les arbres de décision. Ils en résultent des moyennes de plus de 80% d'instances non voyellés correctement classées. Certains attributs morphologiques donnent les mêmes résultats de classification. Ceci peut être expliqué par le fait que les attributs morphologiques associés fournissent peu de nombres de valeurs de classe (ne dépassant pas 6 chacune). D'un autre côté, l'attribut « PRONOM » (par exemple) offre environ 64 valeurs de la classe qui peut générer des résultats distincts pour les différents classifieurs. Parmi les classifieurs possibilistes, nous remarquons que le modèle qui assemble les mesures de possibilité et de nécessité ($\Pi + N$) fournit de meilleurs résultats (87.43%). Ceci confirme la capacité modèle possibiliste à traiter les données imprécises, surtout que les textes arabes ont un taux d'ambiguïté élevé.

Attribut morphologique	Classifieur Bayésien Naïf	Arbres de décision	Classifieur SVM	Classifieur possibiliste utilisant *N*	Classifieur possibiliste utilisant Π	Classifieur possibiliste utilisant Π + N
POS	88.62 %	89.58 %	89.98 %	90.17%	91.58 %	90.45%
ADJECTIF	96.51 %	96.51 %	96.51 %	96.86%	97.58%	97.63%
ASPECT	71.20%	71.20%	71.20%	86.20%	81.78%	86.16%
CAS	56.12 %	56.12%	56.12 %	68.76%	68.40%	76.55%
CONJONCTION	83.03 %	83.03 %	83.03 %	88.66%	95.04%	90.79%
DETERMINANT	64.12%	64.16 %	64.12 %	95.92%	95.25%	96.13%
GENRE	57.15 %	57.15%	57.15 %	90.45%	93.23%	93.78%
MODE	99.32 %	99.32 %	99.32 %	99.96%	99.93%	99.96%
NOMBRE	85.18 %	85.18 %	85.18 %	87.00%	95.30%	93.25%
PARTICULE	96.65 %	96.65%	96.65 %	98.87%	96.91%	98.87%
PERSONNE	60.22 %	60.22 %	60.22 %	65.07%	66.27%	66.88%
PREPOSITION	82.87 %	82.87%	82.87 %	90.20%	88.60%	95.70%
VOIX	71.21 %	71.21 %	71.21 %	78.80%	78.75%	79.05%
PRONOM	55.02 %	55.84 %	56.88 %	59.56%	59.10%	58.79%
Moyenne	**76.23 %**	**76.36 %**	**76.46 %**	**85.46%**	**86.27%**	**87.43%**

TABLEAU 3 : Les taux de désambiguïsation des attributs morphologiques en utilisant 6 classifieurs possibilistes et non-possibilistes dans le corpus du Hadith.

Nous essayons de prouver l'indépendance du domaine de nos modèles possibilistes. Pour ce faire, nous menons nos expérimentations sur le corpus Arabic Treebank rassemblant les textes de journaux. Ce corpus donne les résultats de désambiguïsation de l'attribut POS. A cet effet, les instances des ensembles d'apprentissage et test seront décrites par les attributs POS-2, POS-1, POS+1 et POS+2 qui représentent, respectivement, les catégories grammaticales des deux mots qui suivent et des deux mots qui précédent le mot courant.

Le tableau 4 présente les taux de désambiguïsation de l'attribut POS pour les deux corpus « Hadith » et « Arabic Treebank » données par les six classifieurs.

	Classifieur Bayésien Naïf	Arbres de décision	Classifieur SVM	Classifieur possibiliste utilisant *N*	Classifieur possibiliste utilisant Π	Classifieur possibiliste utilisant Π + N
HADITH	88.62 %	89.58 %	89.98 %	90.17%	91.58 %	90.45%
TREEBANK	80.98%	81.85%	81.77%	83.26%	84.23%	83.35%

TABLEAU 4 : Les taux de désambiguïsation de la catégorie grammaticale des deux coprus « Hadith » et «Arabic Treebank»

Nous obtenons des résultats proches avec des taux élevés. Ces résultats révèlent que l'approche de désambiguïsation possibiliste est indépendante du domaine et de type du texte. Elle fournit des taux raisonnables (plus de 80%) pour les textes de journaux ainsi que pour les textes de Hadith. Il y a, cependant, une différence d'environ 7% entre les deux corpus. Comme les tailles des deux corpus sont presque égales, nous pouvons expliquer ce fait par la nature de l'analyseur morphologique (i.e. *Aramorph*) dont le lexique est plutôt classique. Ainsi, cet outil est incapable d'analyser certaines entrées modernes. Par ailleurs, le corpus du Hadith contient des expressions récurrentes, qui existent à la fois dans les ensembles d'apprentissage et de test (par exemple "صلى الله عليه وسلم" ; Paix et la Bénédiction soient Sur Lui).

Conclusion et perspectives

Nous avons présenté, dans cet article, une nouvelle approche possibiliste pour désambiguïser les attributs morphologiques des textes arabes non voyellés. La désambiguïsation est considérée comme une tâche de classification. A cet égard, nous avons défini un classifieur possibiliste pour apprendre et tester des données imprécises. Nous avons établi trois modèles de classification qui calculent, respectivement, les mesures de possibilité, de nécessité et la somme de ces deux mesures. Nous avons effectué une étude comparative de ces trois modèles de classification possibiliste avec des classifieurs non-possibilistes pour désambiguïser 14 attributs morphologiques. En comparant les résultats des différents classifieurs, nous avons conclu que la théorie possibiliste a donné de meilleurs taux de désambiguïsation quand elle combine les mesures de nécessité et de possibilité.

Malgré ces résultats encourageants, nous avons remarqué que notre approche n'arrive pas à désambiguïser intégralement la totalité des attributs morphologiques. Cela peut s'expliquer par un phénomène linguistique connu en langue Arabe qui se traduit par un ordre relativement aléatoire des mots dans la phrase (Keskes et al., 2013) et également par l'incapacité de désambiguïser les particules qui ont un taux d'ambiguïté élevé, même dans les textes voyellés. Comme perspectives, nous envisageons de faire face à ces problèmes en adoptant l'une des deux alternatives. D'une part, nous pouvons agrandir l'ensemble d'apprentissage. D'autre part, l'intégration d'une analyse linguistique manuelle dans la phase d'apprentissage permettra de filtrer les mots vides et de minimiser le taux d'ambiguïté résultant. Cependant, nous essaierons de réduire le taux d'intervention, pour éviter de traiter tout l'ensemble d'apprentissage à la main. Nous visons aussi à intégrer notre classifieur dans une application de recherche d'information qui traite des textes voyellés et non voyellés, en introduisant une phase primitive de désambiguïsation des requêtes et des documents. A cette étape, nous pouvons renoncer à la désambiguïsation des particules car elles sont considérées comme des mots vides et ne sont pas utilisés pour l'indexation. En outre, les attributs morphologiques calculés par nos outils sont utiles même pour d'autres niveaux d'analyse à savoir syntaxiques et sémantiques (Bounhas et Slimani, 2009).

Références

AYED R., BOUNHAS I., ELAYEB B., EVRARD F., BENLLAMINE BEN SAOUD N. (2012a). A Possibilistic Approach for the Automatic Morphological Disambiguation of Arabic Texts. *In: T. Hochin & R. Lee (Eds.), Proceedings of the 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel Distributed Computing (SNPD)*, Kyoto, Japan,187-194.

AYED R., BOUNHAS I., ELAYEB B., EVRARD F., BENLLAMINE BEN SAOUD N. (2012b). Arabic Morphological Analysis and Disambiguation Using a Possibilistic Classifier. *In Intelligent Computing Theories and Applications, Proceedings of the 8th International Conference on Intelligent Computing (ICIC)*, China, 274-279.

BOUNHAS I., SLIMANI Y. (2009). A hybrid approach for Arabic multi-word term extraction. *In Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE)*, Dalian, China, 1–8.

BOUNHAS I., ELAYEB B., EVRARD F., SLIMANI Y. (2011). Organizing Contextual Knowledge for Arabic Text Disambiguation and Terminology Extraction. *Knowledge Organization* 38(6):473–490.

COLLINS M. (2002). Discriminative training methods for hidden Markov models: theory and experiments with perceptron algorithms. *In Proceedings of the ACL-2nd conference on Empirical methods in natural language processing*, Stroudsburg, PA, USA, 1-8.

DAOUD D. (2009). Synchronized Morphological and Syntactic Disambiguation for Arabic. *Advances in Computational Linguistics* 41, 73-86.

DIAB M., HACIOGLU K., JURAFSKY D. (2004). Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks. *In Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting*, Boston, USA, 149-152.

DUBOID D., PRADE H. (1985). Théorie des possibilités: applications à la représentation des connaissances en informatique. Masson, Paris, France.

DUBOID D., PRADE H. (1994). Possibility Theory: An Approach to computerized Processing of Uncertainty. Plenum Press, New York, USA.

ELAYEB B., EVRARD F., ZAGHDOUD M., BEN AHMED M. (2009). Towards an intelligent possibilistic web information retrieval using multiagent system. *Interactive Technology and Smart Education* 6(1): 40–59.

ELHADJ Y., AL-SUGHAYEIR I., AL-ANSARI A. (2009). Arabic Part-Of-Speech Tagging using the Sentence Structure. *In: Proceedings of the Second International Conference on Arabic Language Resources and Tools*, Cairo, Egypt, 241-245.

ELSHAFEI M., AL-MUHTASEB H., AL-GHAMDI M. (2002). Techniques for high quality Arabic speech synthesis. *Information Sciences 140*(3), 255-267.

FETTWEIS G., MEYR H. (1991). High-speed parallel Viterbi decoding: algorithm and VLSI-architecture. *IEEE Communications Magazine*, 46- 55.

FORNEY G.D. (1973). The Viterbi algorithm. *Proceedings of IEEE* 61: 268-278.

GARSIDE R., LEECH F. (1987). The UCREL probabilistic parsing System. *The Computational Analysis of English: A Corpus-Based Approach*, Longman, London, 66-81.

HABASH N., RAMBOW O. (2005). Arabic Tokenization, Part-of-speech Tagging and Morphological Disambiguation in One Fell Swoop. *In: the proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Stroudsburg*, PA, USA, 573–580.

HABASH N., RAMBOW O. (2007). Arabic Diacritization Through Full Morphological Tagging. *In: Human Language Technologies: The Conference of the North American Chapter of the Association for Computational Linguistics, Stroudsburg*, PA, USA, 53-56.

HABASH N., RAMBOW O., ROTH R. (2009). Mada+tokan: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. *In Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR).*

HAJIC J. (2000). Morphological Tagging: Data vs. Dictionaries. *In: Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference, Stroudsburg*, PA, USA, 94-101.

HAOUARI B., BEN AMOR N., ELOUEDI Z., MELLOULI K. (2009). Naïve possibilistic network classifiers. *Fuzzy Sets and Systems* 160(22): 3224-3238.

HARRAG F., ALOTHAIM A., ABANMY A., ALOMAIGAN F., ALSALEHI S. (2013). Ontology Extraction Approach for Prophetic Narration (Hadith) using Association Rules. *International Journal on Islamic Applications in Computer Science And Technology* 1(2): 48-57.

HOCEINI Y., CHERAGUI M. A., ABBAS M. (2011). Towards a New Approach for Disambiguation in NLP by Multiple Criterian Decision-Aid. *The Prague Bulletin of Mathematical Linguistics 95*, 19-32.

JURAFSKY D., MARTIN J.H. (2009). Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition. *Pearson Prentice Hall, Upper Saddle River*, New Jersey, USA.

KESKES I., BEANAMARA F., HADRICH BELGUITH L. (2013). Segmentation de textes arabes en unités discursives minimales. *TALN-RECITAL, Les sables d'Olonne,* 435-449.

KHOJA SH. (2001). APT: Arabic part-of-speech tagger. *In: Proceedings of Student Workshop at the Second Meeting of the North American Association for Computational Linguistics*, Carnegie Mellon University, Pennsylvania, USA.

KOHAVI R. (1995). A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection. *In: Proceedings of the 14th International Joint Conference on Artificial Intelligence,* San Francisco, CA, USA, 1137-1143.

LIN J., VITTER S, J., HELLERSTEIN L. (1994). A Theory for Memory-Based Learning. *Machine Learning* 17(2-3): 143-167.

MAAMOURI M., BIES A., KULICK S. (2009). Creating a Methodology for Large-Scale Correction of Treebank Annotation: The Case of the Arabic Treebank. *In the proceedings of MEDAR Second International Conference on Arabic Language Resources and Tools*, Cairo, Egypt, 138–144.

MANSOUR S., SIMA'AN K., WINTER Y. (2007). Smoothing a lexicon-based pos tagger for Arabic and Hebrew. *ACL07 Workshop on Computational Approaches to Semitic Languages*, Prague, Czech, 97-103.

OTHMAN E., SHAALAN K., RAFEA A. (2004). Towards Resolving Ambiguity in Understanding Arabic Sentence. *In the proceedings of International Conference on Arabic Language Resources and Tools, NEMLAR*, Egypt, 118-122.

PEARL J. (1988). Probabilistic reasoning in intelligent systems: networks of plausible inference. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmman, San Francisco, California, USA.

QUINLAN J. R. (1986). Induction of decision trees. *Machine Learning* 1(1): 81-106.

ROTH R., RAMBOW O., HABASH N., DIAB M., RUDIN C. (2008). Arabic Morphological Tagging, Diacritization, and Lemmatization Using Lexeme Models and Feature Ranking. *In: Proceedings of the Association for Computational Linguistics conference (ACL)*, Columbus, Ohio, USA, 117-120.

SCHMID H. (1994). Probabilistic part-of-speech tagging using decision trees. *In Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK, 44-49.

TLILI-GUIASSA Y. (2006). Hybrid Method for Tagging Arabic Text. *Journal of Computer Science* 2(3): 245-248.

VAPNIK V. (1998). Statistical Learning Theory. Wiley, New York, USA, 1-736.

ZRIBI C., TORJMEN A., BEN AHMED M. (2006). An Efficient Multi-agent System Combining POS-Taggers for Arabic Texts. *In Proceedings of 7th international conference of Computational Linguistics and Intelligent Text Processing*, LNCS Volume 3878, Springer, 121-131.

Un analyseur discriminant de la famille LR pour l'analyse en constituants

Benoît Crabbé
ALPAGE, INRIA, Université Paris Diderot
Place Paul Ricoeur , 75013 Paris
bcrabbe@linguist.univ-paris-diderot.fr

Résumé. On propose un algorithme original d'analyse syntaxique déterministe en constituants pour le langage naturel inspiré de LR (Knuth, 1965). L'algorithme s'appuie sur un modèle d'apprentissage discriminant pour réaliser la désambiguisation (Collins, 2002). On montre que le modèle discriminant permet de capturer plus finement de l'information morphologique présente dans les données, ce qui lui permet d'obtenir des résultats état de l'art en temps comme en exactitude pour l'analyse syntaxique du français.

Abstract. We provide a new weighted parsing algorithm for deterministic context free grammar parsing inspired by LR (Knuth, 1965). The parser is weighted by a discriminative model that allows determinism (Collins, 2002). We show that the discriminative model allows to take advantage of morphological information available in the data, hence allowing to achieve state of the art results both in time and in accurracy for parsing French.

Mots-clés : Analyse guidée par les têtes, analyse LR, temps linéaire, modèle discriminant,inférence approximative.

Keywords: Head driven parsing, LR parsing, linear time, discriminative modelling, approximate inference.

1 Introduction

Cet article présente un algorithme d'analyse syntaxique robuste inspiré des algorithmes LR (Knuth, 1965) et GLR (Tomita, 1985) pour les grammaires de réécritures non contextuelles. L'algorithme est augmenté d'un mécanisme de pondération permettant la désambiguisation. Celui-ci est basé sur l'algorithme du perceptron global (Collins, 2002). L'article montre que cet algorithme est état de l'art en temps comme en correction sur un jeu de données de référence pour le français (Abeillé *et al.*, 2003; Seddah *et al.*, 2013).

L'analyse syntaxique en constituants repose sur l'hypothèse que caractériser la structure d'une phrase de la langue en modélisant la manière dont les mots sont groupés a du sens. De plus il s'agit d'une représentation particulièrement adaptée à la construction compositionnelle du sens des phrases, comme illustré par (Socher *et al.*, 2012). Il reste que les principaux modèles d'analyse probabiliste en constituants sont génératifs et ont été conçus en priorité pour l'analyse de l'anglais (Petrov *et al.*, 2006; Charniak, 2000; Collins, 2003) ou le chinois (Zhang, 2009), c'est-à-dire des langues à morphologie très pauvre. Pour l'analyse de langues à morphologie plus riche, comme le français, on fait l'hypothèse qu'il est souhaitable de pouvoir se donner une représentation structurée des mots (lemmatisation, analyse morphologique, représentation sémantique) issue directement de treebanks voire de ressources exogènes, comme des dictionnaires. Et ce, dans le but de capturer certaines de leurs propriétés essentielles et de combattre les effets de dispersion des données. Il est cependant non trivial d'adapter les modèles génératifs développés en priorité pour l'anglais au cas des langues à morphologie riche. Il faudrait notamment mettre en place une factorisation du modèle génératif relativement lourde impliquant notamment le déploiement de méthodes de lissage très élaborées. Pour cette raison, il semble préférable d'utiliser un modèle d'apprentissage discriminant qui offre naturellement la possibilité de factoriser le modèle d'analyse.

Cependant, comme montré par (Finkel *et al.*, 2008) la conception de modèles d'analyse syntaxique en constituants entièrement discriminants est une tâche difficile qui pose des problèmes notoires d'efficacité. Ainsi la pratique courante pour l'analyse syntaxique en constituants consiste plutôt à utiliser un modèle discriminant pour réordonnancer un sous-ensemble des hypothèses d'analyse construites par un analyseur génératif efficace (Charniak & Johnson, 2005). Dans cet article on montre qu'en introduisant une approximation appropriée, on peut formuler directement un algorithme d'analyse en constituants basé sur un modèle discriminant qui est plus expressif et plus efficace que ses contreparties génératives. On

montre en particulier qu'il permet de tirer parti de l'information morphologique naturellement présente dans les données pour obtenir des résultats état de l'art pour le français en temps comme en correction.

L'article est structuré comme suit. On commence par établir en section 2 un cadre formel approprié qui permet notamment de contraster le problème de l'analyse en constituants avec celui de l'analyse en dépendances. Ayant observé que la structure des arbres de constituants est contrainte, on propose en section 3 une méthode destinée à contraindre l'espace des analyses possibles par un automate LR pour le cas de l'analyse robuste, ce qui distingue notre proposition de (Sagae & Lavie, 2006; Zhang, 2009; Zhu *et al.*, 2013). La section 4 propose un algorithme d'analyse de la famille LR et un algorithme d'apprentissage basé sur le modèle du perceptron global. On y détaille en particulier l'usage d'une méthode d'inférence approximative basée sur un faisceau et ses conséquences sur la méthode d'apprentissage. Les sections 5 et 6 introduisent enfin les extensions de l'algorithme de base qui sont utiles en pratique et qui permettent de structurer les formes lexicales. Finalement la section 7 donne une évaluation quantitative des différents composants de l'algorithme et une comparaison avec l'état de l'art.

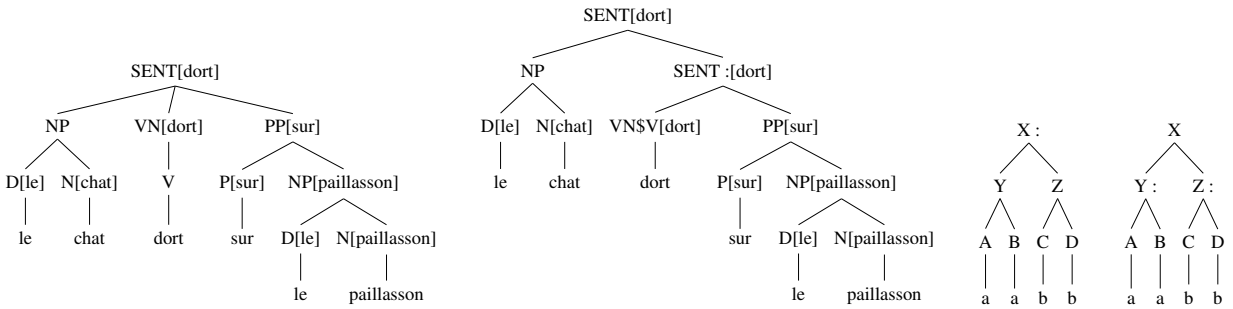
2 Représentation de la grammaire

On pose comme hypothèse de départ que la grammaire manipulée lors de l'analyse est une grammaire lexicalisée de type 2-LCFG (Nederhof & Satta, 2010). Une grammaire 2-LCFG est une grammaire CFG dont les règles ont nécessairement la forme donnée en Table 1 où les symboles h, x dénotent des symboles terminaux. Les symboles de la forme $A[h]$ dénotent des symboles non terminaux lexicalisés. Un tel symbole est composé d'un non terminal délexicalisé noté A, B, C et d'un terminal h ou x. Une règle 2-LCFG sera par exemple de la forme $NP[chat] \rightarrow D[le]\ N[chat]$. Le symbole de la forme $X[h]$ situé en partie droite de la règle est appelé tête de la règle. Une grammaire de type 2-LCFG comporte en pratique un très grand nombre de règles. L'analyse efficace de ce type de grammaire demande typiquement de générer dynamiquement les non terminaux lexicalisés en cours d'analyse. Dans ce qui suit, nous focalisons d'abord sur les propriétés de la grammaire 2-CFG sous jacente. La partie dynamique sera détaillée en Section 4.

$$\begin{array}{lcl} A[h] & \rightarrow & B[h]\,C[x] \\ A[h] & \rightarrow & B[x]\,C[h] \\ A[h] & \rightarrow & h \end{array}$$

TABLE 1 – Formes des règles 2-LCFG

Transformation du treebank Le treebank français, comme la plupart des treebanks existants encode des arbres d'arité n, avec notamment $n > 2$. En supposant un tel treebank, dont les noeuds sont tous annotés par leurs têtes, on effectue les opérations de précompilation suivantes : d'abord une opération de binarisation, appelée Markovisation par la tête d'ordre 0 (Collins, 2003), suivie d'une procédure de réduction des règles unaires internes, comme illustré en Figure 1. Ces deux opérations garantissent que le treebank transformé comporte uniquement des arbres dont la structure suit une forme normale de Chomsky. La procédure de binarisation introduit un ensemble de catégories temporaires que nous notons avec le suffixe ':' en Figure 1. Le symbole '$' dénote les catégories introduites par la réduction de règles unaires.



Arbre source — Arbre 2-LCFG transformé — Arbres 2-CFG invalides

FIGURE 1 – Représentation des arbres de treebank par une 2-LCFG

Propriétés des arbres transformés Comme le treebank est mis en forme normale de Chomsky, on peut montrer par induction que le nombre de pas de dérivation η pour dériver à partir de la grammaire 2-CFG sous-jacente au treebank

tout arbre d'analyse d'une phrase de n mots est constant : $\eta = 2n - 1$. Cette propriété est en principe essentielle (voir également section 5) et explique pourquoi on utilise 2-LCFG pour représenter la grammaire. On remarque de plus que les arbres du treebank transformé ont une structure contrainte. Par exemple la racine d'un arbre ne peut pas être un symbole temporaire ou un arbre ne peut pas avoir deux noeuds temporaires en partie droite d'une même règle. Les analyseurs en dépendances projectifs vérifient également la première propriété : $\eta = 2n - 1$ (Huang & Sagae, 2010). Par contre les structures à analyser ne sont pas contraintes de manière identique.

3 Construction d'un automate LR pour une grammaire de treebank

Les propriétés des arbres transformés, induites par l'introduction des symboles temporaires, ont pour conséquence qu'on ne peut pas construire naïvement un analyseur par décalage réduction non contraint comme c'est le cas pour l'analyse en dépendances. Dans le cas 2-CFG présenté ici, les arbres d'analyse ont une structure contrainte et il s'agit de garantir que l'analyseur produise des arbres qui respectent ces contraintes. Dans son parser, (Sagae & Lavie, 2006) ne gère pas ce problème et retourne à posteriori des analyse partielles dans le cas où l'arbre d'analyse s'avère invalide. (Zhang, 2009) exprime plutôt des contraintes locales dans l'algorithme d'analyse qui sont destinées à empêcher l'analyseur de produire des arbres invalides. Celles-ci comportent notamment des contraintes d'ordre général et des contraintes plus spécifiques à la grammaire du chinois manipulée par l'auteur. On propose ici une solution plus générale qui consiste à garantir la correction des arbres d'analyse par l'utilisation de tables $LR(0)$. Tel quel, un analyseur $LR(0)$ traditionnel (Knuth, 1965) n'est pas approprié pour l'analyse du langage naturel : le principe de cette méthode d'analyse est d'éliminer statiquement l'ambiguité de la grammaire. Dans ce qui suit, à l'instar de (Tomita, 1985) on construit des tables LR sans chercher à éliminer statiquement l'ambiguité. Au contraire, on préserve les conflits dans la table : la désambiguisation est réalisée dynamiquement par un mécanisme de pondération. L'utilisation de tables $LR(0)$ permet plutôt de garantir que l'analyseur produit des arbres corrects et dérivés d'une grammaire bien identifiée. Contraindre la structure des arbres d'analyse revient alors à contraindre la grammaire ayant servi à générer ces tables. Ceci dit, construire un automate $LR(0)$ pour l'analyse robuste à partir de treebank pose à priori deux problèmes. Le premier est inductif : il faut garantir que la grammaire extraite d'un treebank généralise à du texte non vu. Rien ne dit que la grammaire issue d'un échantillon limité est générale. Le second est d'ordre pratique : une grammaire extraite de treebank est généralement une grammaire de très grande taille et massivement ambigue. Cette seconde propriété rend la compilation d'automates $LR(0)$ délicate en pratique : la construction de tables $LR(0)$ fait intervenir un algorithme de déterminisation d'automates dont la complexité est en $\mathcal{O}(2^n)$ avec n le nombre d'états de l'automate LR non déterministe. Dans le cas de très grosses grammaires ambigues, n est très élevé et on est proche du pire des cas.

Pour ces deux raisons, et dans le cas où l'analyseur résultant a pour but d'être robuste, on propose de dériver l'automate à partir d'une construction grammaticale basée sur des classes d'équivalence. On commence par poser que Σ est l'ensemble des symboles non terminaux extraits du treebank tel que T est l'ensemble des symboles temporaires introduits par la procédure de binarisation et N est l'ensemble des autres symboles ($\Sigma = N \cup T$ et $N \cap T = \emptyset$). On note W l'ensemble des symboles terminaux extraits du treebank et $A \in \Sigma$ l'axiome unique de cette grammaire ($A \in N$). On définit ensuite un ensemble de classes d'équivalence qui partitionnent Σ : $[a] = \{A\}$, $[t] = T$ et $[n] = \Sigma - (T \cup \{A\})$. Pour uniformiser les notations on définit $[w] = W$. Munis de ces classes d'équivalence, on définit la grammaire matrice $G_m = \langle \Sigma_m, [w], [a], R_m \rangle$ où $\Sigma_m = \{[a], [t], [n]\}$. Reste à définir R_m pour que les règles respectent les contraintes de bonne formation des arbres. On formule en table 2 un exemple de telles règles au format ID/LP (Gazdar *et al.*, 1985). Autrement dit, une règle de dominance immédiate de la forme $a \rightarrow b\,,\ c$ est expansée en deux règles de réécriture $a \rightarrow b\,c$ et $a \rightarrow c\,b$.

$[a] \rightarrow [n]\,,\ [t]$	$[n] \rightarrow [n]\,,\ [t]$	$[t] \rightarrow [n]\,,\ [t]$
$[a] \rightarrow [n]\,,\ [n]$	$[n] \rightarrow [n]\,,\ [n]$	$[t] \rightarrow [n]\,,\ [n]$
$[a] \rightarrow [w]$	$[n] \rightarrow [w]$	

TABLE 2 – Exemple de règles de dominance immédiate pour G_m

La grammaire G_m implémente les contraintes de bonne formation des arbres mentionnées ci-dessus, comporte en pratique peu de règles et elle est robuste : $L(G_m) = [w]^*$. On peut construire très facilement à partir de G_m un automate $LR(0)$ déterministe $A_m = \langle \Sigma_m \cup \{[w]\}, Q, i, F, E_m \rangle$ en utilisant les méthodes classiques (Aho *et al.*, 2006). On construit l'automate expansé $A_{exp} = \langle \{\Sigma \cup W\}, Q, i, F, E \rangle$ où $E = \{(q, a, q') \,|\, (q, [x], q') \in E_m, \forall a \in [x]\}$. L'automate A_{exp}

peut alors être utilisé comme guide par un analyseur $LR(0)$ non déterministe qui garantit de produire des arbres d'analyse corrects.

Pour construire la table d'analyse LR, il faut encore définir l'ensemble des actions $\mathcal{A}$ de l'analyseur : $\mathcal{A} \stackrel{def}{=} \{RL(X)|X \in \Sigma\} \cup \{RR(X)|X \in \Sigma\} \cup \{RU(X)|X \in \Sigma\} \cup \{S\}$. Il s'agit d'un jeu d'actions analogue à celui de (Sagae & Lavie, 2006). Si les actions sont définies explicitement dans la section suivante, signalons déjà que $RL(X)$ (resp. $RR(X)$) est une action de réduction de règle binaire par le non terminal X tel que le premier (resp. second) élément de la partie droite est assigné comme tête de la règle. $RU(X)$ dénote une réduction unaire par un non terminal X, et S le décalage. Par contraste avec un jeu d'actions LR classique (Aho *et al.*, 2006), celui-ci introduit une approximation : pour un état de l'automate donné, on introduit une action $RL(X)$ et une action $RR(X)$ si il existe un item LR de la forme $\langle X \rightarrow A\, B\bullet \rangle$ dans un état $q_i \in A_{exp}$ sans exiger que $\langle X \rightarrow B\, A\bullet \rangle \in q_i$. Cette simplification permet de réduire le nombre d'actions de l'analyseur, ce qui facilite l'apprentissage du modèle et a également un effet optimisant (Section 4).

Si la grammaire matrice G_m présentée ici n'est pas la seule grammaire matrice possible (voir également Figure 7 pour l'expression de sous-grammaires locales), un jeu de règles R_m valide pour G_m doit au moins implémenter les contraintes de bonne formation des arbres mentionnées ci-dessus en s'appuyant sur une partition de Σ en classes d'équivalence. Par contre le jeu d'actions simplifié utilisé ici suppose que pour toute règle de grammaire $R \in R_m$ de la forme $[x] \rightarrow [y]\,[z]$, on a également une règle de grammaire $R' \in R_m$ de la forme $[x] \rightarrow [z]\,[y]$. C'est pour cette raison que nous formulons les règles R_m au format ID/LP et cela signifie qu'on ne peut imposer de contraintes dures sur l'ordre des mots dans la grammaire, comme c'est le cas dans la plupart des analyseurs syntaxiques robustes.

4 Algorithme d'analyse inspiré de LR et pondéré par un perceptron

Bien qu'inspiré de LR, l'algorithme d'analyse proposé est un algorithme naturellement non déterministe. Le déterminisme est apporté par un système de pondérations basé sur l'algorithme du perceptron global (Collins, 2002). On commence par présenter l'algorithme d'analyse pondéré avant de décrire la méthode d'estimation des poids du perceptron à partir d'un treebank.

Algorithme d'analyse On suppose que l'algorithme analyse des séquences de tokens $\mathcal{T} = t_1 \ldots t_n$ et qu'une table d'analyse $LR(0)$ a été construite. La fonction GOTO de cette table est la fonction GOTO :$(\Sigma \cup W) \times \mathbb{N} \mapsto \mathbb{N}$ qui envoie des couples de symbole et d'état vers un nouvel état de l'automate LR. La fonction ACTION de cette table est la fonction ACTION :$(\mathbb{N} \times W) \mapsto 2^{\mathcal{A}}$ qui retourne l'ensemble **a** des actions possibles étant donné un couple de terminal et de numéro d'état. On note σ_i l'état initial de l'automate LR et σ_e un état final.

Les algorithmes à décalage réduction manipulent habituellement une pile et une liste d'attente. Ici, l'algorithme manipule explicitement une pile et implicitement une liste d'attente. La pile $\mathbf{S} = \ldots|s_2|s_1|s_0$, de sommet s_0, est faite de noeuds de la forme $s_i = \langle \sigma, \tau \rangle$ où σ est un numéro d'état LR et $\tau = (s_i.c_t[s_i.w_t]\; s_i.c_l[s_i.w_l]\; s_i.c_r[s_i.w_r])$ dénote un arbre local de profondeur 1. $s_i.c_t, s_i.c_l, s_i.c_r$ dénotent respectivement les catégories du noeud racine, de son fils gauche et de son fils droit. $s_i.w_t, s_i.w_l, s_i.w_r$ dénotent respectivement les items lexicaux du noeud racine, de son fils gauche et de son fils droit. La notation $s_i.c.[s_i.w.]$ encode donc un symbole non terminal d'une 2-LCFG au noeud s_i de la pile d'analyse (Figure 2).

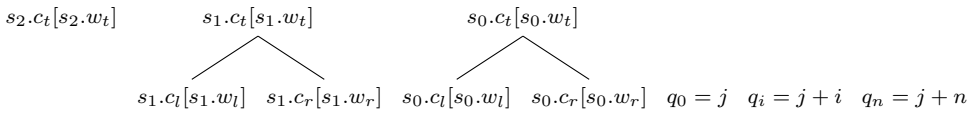


FIGURE 2 – *Représentation graphique du vecteur d'accroches* $\boldsymbol{\kappa}$ Le vecteur $\boldsymbol{\kappa}$ représente l'information localement accessible pour valuer les traits $\phi_i(a, \boldsymbol{\kappa}, j)$. On représente graphiquement de droite à gauche les différents noeuds de la pile d'exécution. Les symboles $s_i.w_{(.)}$ encodent les formes lexicales têtes des constituants et les symboles $s_i.c_{(.)}$ encodent les catégories des constituants. De plus, les symboles q_i encodent les mots de la file d'attente. On remarque que connaître la valeur de l'index j permet d'addresser non seulement le premier mot de la file d'attente mais également les suivants.

L'algorithme d'analyse fonctionne en empilant et en dépilant des noeuds de la pile **S** et en défilant progressivement la file d'attente. Ainsi la configuration courante ou état de l'analyseur est un couple $C_i = \langle j, \mathbf{S} \rangle$ qui dénote la position courante

ITEM	$\langle j, \mathbf{S}\rangle : w$
INIT	$\langle 1, \langle \sigma_i, \epsilon\rangle\rangle : 0, \emptyset$
GOAL	$\langle n+1, \langle \sigma_e, \mathbf{S}\rangle\rangle : w$

SHIFT $\dfrac{\langle j,\mathbf{S}_\ominus \mid s_0{=}\langle \sigma,\_\rangle\rangle : w}{\langle j{+}1,\mathbf{S}_\ominus \mid s_0 \mid \langle \textsc{Goto}(t_j,\sigma),(t_j[j]\_\_)\rangle : w{+}F(S,\langle j,\mathbf{S}\rangle)}$

RL(X) $\dfrac{\langle j,\mathbf{S}_\ominus \mid s_2{=}\langle \sigma_2,\_\rangle{:}w_2 \mid s_1{=}\langle \sigma_1,(s_1.c_t[s_1.w_t]\_\_)\rangle{:}w_1 \mid s_0{=}\langle \sigma_0,(s_0.c_t[s_0.w_t]\_\_)\rangle\rangle{:}w_0}{\langle j,\mathbf{S}_\ominus \mid s_2 \mid \langle \textsc{Goto}(X,\sigma_2),(X[s_1.w_t]\ s_1.c_t[s_1.w_t]\ s_0.c_t[s_0.w_t])\rangle{:}w_0{+}F(RL(X),\langle j,\mathbf{S}\rangle)}$

RR(X) $\dfrac{\langle j,\mathbf{S}_\ominus \mid s_2{=}\langle \sigma_2,\_\rangle{:}w_2 \mid s_1{=}\langle \sigma_1,(s_1.c_t[s_1.w_t]\_\_)\rangle{:}w_1 \mid s_0{=}\langle \sigma_0,(s_0.c_t[s_0.w_t]\_\_)\rangle\rangle{:}w_0}{\langle j,\mathbf{S}_\ominus \mid s_2 \mid \langle \textsc{Goto}(X,\sigma_2),(X[s_0.w_t]\ s_1.c_t[s_1.w_t]\ s_0.c_t[s_0.w_t])\rangle{:}w_0{+}F(RR(X),\langle j,\mathbf{S}\rangle)}$

RU(X) $\dfrac{\langle j,\mathbf{S}_\ominus \mid s_1{=}\langle \sigma_1,(s_1.c_t[s_1.w_t]\_\_)\rangle \mid s_0{=}\langle \sigma_0,(s_0.c_t[s_0.w_t]\_\_)\rangle\rangle{:}w_0}{\langle j,\mathbf{S}_\ominus \mid s_1 \mid \langle \textsc{Goto}(X,\sigma_1),(X[s_0.w_t]\ s_0.c_t[s_0.w_t])\rangle{:}w_0{+}F(RU(X),\langle j,\mathbf{S}\rangle)}$

GR $\dfrac{\langle j,\mathbf{S}_\ominus \mid s_1{=}\langle \sigma_1,(s_1.c_t[s_1.w_t]\_\_)\rangle \mid s_0{=}\langle \sigma_0,(s_0.c_t[s_0.w_t]\_\_)\rangle\rangle{:}w_0}{\langle j,\mathbf{S}_\ominus \mid s_1 \mid \langle \textsc{Goto}(GR,\sigma_1),(s_0.c_t[s_0.w_t]\_\_)\rangle{:}w_0{+}F(GR,\langle j,\mathbf{S}\rangle)}$ (Règle introduite en section 5)

FIGURE 3 – Règles d'inférence de l'algorithme en notation déductive étendue

dans la file d'attente et l'état courant de la pile. Étant donnée une configuration initiale $C_0 = \langle 1, \langle \sigma_i, \epsilon\rangle\rangle$, l'analyseur dérive pas à pas de nouvelles configurations $C_i = \langle j', \mathbf{S}'\rangle$ à partir de configurations $C_{i-1} = \langle j, \mathbf{S}_\ominus | \langle \sigma, \tau\rangle\rangle$ en exécutant une action $a_{i-1} \in \textsc{Action}(\sigma, t_j)$, ce que l'on note $C_{i-1} \overset{a_{i-1}}{\Rightarrow} C_i$. Une dérivation de $k-$pas est la séquence $C_{0\Rightarrow k}$ telle que $C_0 \overset{a_0}{\Rightarrow} \ldots \overset{a_{k-1}}{\Rightarrow} C_k$. Une dérivation est terminée dans deux cas. La configuration $C_{3n-1} = \langle n+1, \langle \sigma, \tau\rangle\rangle$ est générée. Si $\sigma = \sigma_e$ la dérivation est un succès. Une dérivation est également terminée lorsque $\textsc{Action}(\sigma, t_j) = \emptyset$ pour une configuration $C_k = \langle j, \mathbf{S}_\ominus | \langle \sigma, \tau\rangle\rangle$ donnée, c'est le cas d'échec. Les actions exécutées en cours de dérivation modifient la pile d'analyse et l'état d'avancement j sur la liste d'attente. Celles-ci sont définies en Figure 3 en notation déductive étendue.

Une dérivation $C_{0\Rightarrow k} = C_0 \overset{a_0}{\Rightarrow} \ldots \overset{a_{k-1}}{\Rightarrow} C_k$ est également pondérée par une fonction de la forme :

$$W(C_{0\Rightarrow k}) = \mathbf{w} \cdot \mathbf{\Phi}_g(C_{0\Rightarrow k}) = \sum_{i=0}^{k-1} \mathbf{w} \cdot \mathbf{\Phi}(a_i, C_i) = \sum_{i=0}^{k-1} F(a_i, C_i) \tag{1}$$

où $\mathbf{w} \in \mathbb{R}^d$ est un vecteur d-dimensionnel de poids et $\mathbf{\Phi}(a_i, C_i) \in \{0,1\}^d$ est un vecteur d-dimensionnel de traits. Chaque dimension $0 \leq i \leq d$ du vecteur de traits est valuée par une fonction ϕ_i de signature $\phi_i(a, \boldsymbol{\kappa}, j)$ où $\boldsymbol{\kappa}$ est une séquence d'accroches extraite localement parmi les trois éléments supérieurs de la pile comme illustré en Figure 2. Comme la fonction de pondération est décomposée en une somme de termes correspondant aux étapes de la dérivation, le calcul des poids est réalisé dynamiquement en cours d'analyse et est associé à une configuration (Figure 3) de telle sorte que dans le cas pondéré une configuration a la forme étendue $C_k = \langle j, \mathbf{S}\rangle : w$ où $w = W(C_{0\Rightarrow k})$ est le score préfixe de $C_{0\Rightarrow k}$.

Le système de poids permet de modéliser la désambiguisation en autorisant de choisir l'analyse de poids le plus élevé parmi l'ensemble des analyses possibles pour une séquence $\mathcal{T} = t_1 \ldots t_n$ de mots. Dans notre cas, le non déterminisme est introduit dans l'algorithme par la fonction $\textsc{Action}(\sigma, t_j)$ qui renvoie un ensemble $\mathbf{a} \subseteq \mathcal{A}$ d'actions possibles à effectuer étant donné la configuration courante de l'analyseur de telle sorte qu'à partir d'une séquence $C_{0\Rightarrow k-1}$ on peut dériver un ensemble de séquences $\delta(C_{0\Rightarrow k-1}) = \{C_{0\Rightarrow k} | C_{0\Rightarrow k-1} \overset{a_{k-1}}{\Rightarrow} C_{0\Rightarrow k}\}$ à l'étape suivante. Si $\textsc{Gen}_k(\mathcal{T})$ est l'ensemble des dérivations de $k-$pas pour $\mathcal{T}$, trouver la meilleure analyse demande de calculer la solution du problème d'optimisation suivant [1] :

$$\hat{\mathcal{C}} = \underset{C_{0\Rightarrow 3n-1} \in \textsc{Gen}_{3n-1}(\mathcal{T})}{\text{argmax}} W(C_{0\Rightarrow 3n-1}) \tag{2}$$

On peut observer que le nombre de solutions possibles $|\textsc{Gen}_{3n-1}(\mathcal{T})|$ est de l'ordre de $|\mathcal{A}|^{3n-1}$. Autrement dit, l'espace des solutions est de taille exponentielle. Si il est théoriquement possible de calculer en théorie une solution optimale à ce problème par programmation dynamique en temps polynomial dans la lignée de (Tomita, 1985), le mécanisme

1. Notons que le nombre de pas dérivation $\eta = 2n - 1 + n = 3n - 1$ car dans le cas d'analyse par décalage réduction, il faut réaliser n décalages en plus des $2n - 1$ étapes induites par la forme normale de Chomsky.

de pondération utilisé ici demande également d'évaluer des produits scalaires tels que donné en équation (1) pour des vecteurs qui en pratique sont de très haute dimensionnalité. Le calcul dynamique de poids est en pratique très coûteux en temps et rend très difficile la recherche d'une solution à la fois optimale et efficace au problème d'optimisation, même par programmation dynamique. Dans ce qui suit, nous sacrifions la recherche de la solution optimale pour privilégier l'efficacité en faisant appel à une approximation par faisceau. Étant donné un faisceau $\textsc{Gen}^K_{k-1}(\mathcal{T})$ de taille K, et $\Delta(\textsc{Gen}^K_{k-1}(\mathcal{T})) = \bigcup_{C_{0\Rightarrow k-1}\in\textsc{Gen}^K_{k-1}(\mathcal{T})} \delta(C_{0\Rightarrow k-1})$ l'ensemble des configurations dérivables à partir de ce faisceau, on construit récursivement $\textsc{Gen}^K_k(\mathcal{T})$ comme suit :

$$\textsc{Gen}^K_k(\mathcal{T}) = \underset{C_{0\Rightarrow k}\in\Delta(\textsc{Gen}^K_{k-1}(\mathcal{T}))}{\text{K-argmax}} W(C_{0\Rightarrow k}) \tag{3}$$

Autrement dit, un faisceau ne prend en compte que les $K-$meilleures hypothèses de l'étape de calcul précédente pour réaliser les calculs à l'étape courante. Introduire un faisceau apporte un gain d'efficacité qui rend l'algorithme d'analyse utilisable en pratique. La complexité en temps de l'analyse est d'ordre linéaire : $\mathcal{O}(K|\mathcal{A}|(3n-1)) = \mathcal{O}(n)$. L'introduction d'un faisceau a deux contreparties immédiates. D'une part la solution au problème d'analyse donné en équation (2) n'est plus nécessairement optimale. D'autre part on peut montrer que le faisceau sacrifie également la complétude de l'algorithme. Dans certains cas, l'algorithme peut manquer de trouver la moindre analyse alors qu'il est en fait possible d'en trouver au moins une : c'est le cas où l'ensemble des dérivations menant à une solution valide sont tout simplement élaguées prématurément par le faisceau. En résumé, l'algorithme d'analyse présenté ici est une fonction de prédiction d'arbre qui remplace le calcul optimal donné en équation (2) par l'approximation suivante :

$$\tilde{\mathcal{C}} = \underset{C_{0\Rightarrow 3n-1}\in\textsc{Gen}^K_{3n-1}(\mathcal{T})}{\text{argmax}} W(C_{0\Rightarrow 3n-1}) \tag{4}$$

En pratique, le calcul dynamique des poids est le facteur limitant en temps. Trois aspects propres à l'implémentation sont à signaler, notamment pour obtenir un temps d'analyse linéaire en pratique. En premier lieu, l'ensemble des configurations C_k générées en cours d'analyse est implémenté par une structure de données, commune à toutes les hypothèses concurrentes, appelée pile structurée en arbre (*TSS*), initialement introduite par (Tomita, 1988). Cette structure de données permet d'éviter la copie intempestive de piles d'analyse potentiellement profondes en les encodant dans une structure d'arbre.

En second lieu, nous utilisons systématiquement un noyau de hachage (*hash trick*). Un noyau de hachage (Shi *et al.*, 2009) est une technique d'implémentation dont le but premier est d'accélérer l'exécution de produits scalaires. En supposant une fonction de hachage $h : I \mapsto \{1 \dots D\}$ qui envoie l'index $i \in I$ de chaque fonction ϕ_i sur sa valeur hachée $h(i) = k$, de telle sorte que $\bar{\phi}_k(a, \boldsymbol{\kappa}, j) = \phi_i(a, \boldsymbol{\kappa}, j)$. Nous utilisons donc la fonction de pondération suivante plutôt que la première version donnée en équation 1 :

$$F(a_i, C_i) = \mathbf{w} \cdot \bar{\boldsymbol{\Phi}}(a_i, C_i) \tag{5}$$

où $\bar{\boldsymbol{\Phi}} = \bar{\phi}_1 \dots \bar{\phi}_D$ (avec en pratique $D = 2 \times 10^7 - 7$). Autrement dit, l'utilisation d'un noyau de hachage approxime une fonction ϕ_i par sa fonction image $\bar{\phi}_k$ sans chercher à résoudre les collisions éventuelles de hachage. Cette technique prend son sens lorsqu'on sait que les vecteurs $\boldsymbol{\Phi}$ sont des vecteurs creux à très haute dimensionnalité et qu'ils sont habituellement implémentés par des tables de hachage. Or la résolution des collisions dans ces tables est en pratique très coûteuse en temps. Outre le gain en efficacité apporté par cette technique, le noyau de hachage a deux effets de bords utiles : il permet une réduction de la dimensionalité des vecteurs $\boldsymbol{\Phi}$ et $\mathbf{w}$ et il permet dans une certaine mesure de réduire les effets de surentrainement.

En troisième lieu, nous utilisons une méthode de mémoïsation pour l'évaluation des scores. Celle-ci repose sur une décomposition de la fonction F qui remplace la version donnée en équation 5 par sa version finale :

$$F(a_i, C_i) = \mathbf{w} \cdot \boldsymbol{\Phi}(a_i, C_i) + \sum_{m=1}^{M} \boldsymbol{\mu}_m(a_i, C_i) \tag{6}$$

où $\boldsymbol{\mu}_m$ sont des mémo-fonctions. Lorsqu'elle est évaluée la première fois, une mémo fonction mémorise et renvoie la valeur $\mathbf{w} \cdot \boldsymbol{\Phi}_m(a_i, C_i)$. Si elle est évaluée ultérieurement pour un couple (a'_i, C'_i) équivalent, elle renvoie la valeur mémorisée. Le vecteur de traits $\boldsymbol{\Phi}_m = \phi_1 \dots \phi_r$ d'une mémo fonction $\boldsymbol{\mu}_m$ est constitué de fonctions de signatures $\zeta_m(a, \boldsymbol{\kappa}, j)$ identiques. Deux couples $(a_i, C_i), (a'_i, C'_i)$ sont considérés équivalents par la mémo fonction si la valuation qu'ils donnent aux paramètres $a, \boldsymbol{\kappa}, j$ de ζ_m sont identiques. L'analyseur décrit dans cet article utilise deux mémo-fonctions : $\boldsymbol{\mu}_{lex}$ a pour signature $\zeta_{lex}(a, s_0.w_t, s_0.w_l, s_0.w_r, s_1.w_t, s_1.w_l, s_1.w_r, s_2.w_t, j)$ et $\boldsymbol{\mu}_{cat}$ a pour signature $\zeta_{cat}(a, s_0.w_t, s_0.c_t, s_1.w_t, s_1.c_t, j)$. Remarquons également que les traits sont partitionnés dans des ensembles disjoints : chaque mémo-fonction est responsable de l'évaluation d'un sous-ensemble des traits. Les fonctions ϕ_i dont la signature est incompatible avec toutes les mémo-fonctions existantes forment le vecteur $\boldsymbol{\Phi}$ restant (Equation 6) dont le produit scalaire correspondant est réévalué systématiquement.

Estimation des poids La partie prédictive du modèle d'analyse que nous avons décrite jusqu'à présent est en principe valable pour une famille de modèles linéaires multiclasses comme des modèles de type SVM structuré ou des modèles de champs conditionnels aléatoires. La méthode d'estimation des poids décrite ici est spécifique à l'algorithme du perceptron structuré (Collins, 2002). Ce dernier algorithme a en effet l'avantage d'offrir une méthode d'estimation des poids plus efficace et plus simple à mettre en oeuvre pour le cas de l'analyse syntaxique par comparaison avec les autres modèles linéaires que nous connaissons.

On se penche maintenant sur le problème d'estimation du vecteur de poids $\mathbf{w}$ à partir d'un treebank. La méthode d'estimation des poids pour le perceptron structuré est une généralisation de la méthode du percepron multiclasse et se formule comme suit. On suppose un jeu de données d'entrainement de N phrases, $T = ((\mathcal{T}_1, \mathcal{R}_1), \ldots, (\mathcal{T}_N, \mathcal{R}_N))$ pour lesquelles nous disposons de l'analyse correcte $\mathcal{R}_i$. L'algorithme du perceptron itère sur les données un nombre prédéterminé d'époques E. Pour chaque exemple, l'analyseur compare la meilleure analyse qu'il prédit $\hat{\mathcal{C}}$, étant donnée la valeur courante du vecteur $\mathbf{w}$ de poids, avec l'analyse correcte $\mathcal{R}_i$. En cas d'erreur, l'algorithme met à jour les poids en pénalisant les poids associés à l'analyse prédite et en favorisant les poids associés à l'analyse correcte :

1: **function** PERCEPTRONLEARN($\mathbf{w}, (\mathcal{T}_1, \mathcal{R}_1) \ldots (\mathcal{T}_N, \mathcal{R}_N), E$)
2: $\quad \mathbf{w} \leftarrow 0^d$
3: $\quad \bar{\mathbf{w}} \leftarrow 0^d$
4: $\quad$ **for** $e = 1$ **to** E **do** ▷ Iterations sur E epoch
5: $\quad\quad$ **for** $i = 1$ **to** N **do** ▷ Iterations sur les données
6: $\quad\quad\quad \hat{\mathcal{C}} = \text{argmax}_{C_{0\Rightarrow 3n-1} \in \text{GEN}_{3n-1}(\mathcal{T}_i)}\ \mathbf{w} \cdot \mathbf{\Phi}_g(C_{0\Rightarrow 3n-1})$ ▷ Prédiction (analyse de le phrase)
7: $\quad\quad\quad$ **if** $\hat{\mathcal{C}} \neq \mathcal{R}_i$ **then** ▷ Mise à jour
8: $\quad\quad\quad\quad \mathbf{w} \leftarrow \mathbf{w} + \Phi_g(\mathcal{R}_i) - \Phi_g(\hat{\mathcal{C}})$
9: $\quad\quad\quad \bar{\mathbf{w}} \leftarrow \bar{\mathbf{w}} + \mathbf{w}$ ▷ Moyennage
10: $\quad$ **return** $\bar{\mathbf{w}}/(N \times E)$

L'algorithme du perceptron n'offre pas de garantie de convergence si les données ne sont pas linéairement séparables, ce qui est généralement le cas pour l'analyse syntaxique. Toutefois, en supposant que l'algorithme génère en cours d'analyse différents vecteurs de poids $\mathbf{w}^{(1)} \ldots \mathbf{w}^{(N \times E)}$ proches de la solution optimale du problème d'optimisation, nous utilisons un algorithme du perceptron moyenné dont l'estimation finale des poids est la moyenne de l'ensemble des vecteurs de poids générés en cours d'apprentissage (Collins, 2002).

Revenons sur l'usage du faisceau dans le cas de l'apprentissage structuré avec l'algorithme du perceptron. L'algorithme du perceptron fait l'hypothèse que chaque étape de prédiction renvoie l'analyse de poids le plus élevé pour un exemple donné (ligne 6), en utilisant l'équation (2). Or l'usage d'une méthode de prédiction approximative avec faisceau ne permet pas de garantir cette hypothèse. L'approximation par faisceau renvoie la solution à l'équation (4) et il n'est pas garanti que $\tilde{\mathcal{C}} = \hat{\mathcal{C}}$. Pire encore, il existe des cas où $\tilde{C} \neq \mathcal{R}$ alors que $\hat{C} = \mathcal{R}$: ce sont les cas où l'hypothèse $\hat{\mathcal{C}}$ n'est pas générée car écartée prématurément par le faisceau. Autrement dit, l'approximation introduite par le faisceau peut causer une mise à jour des poids invalide et par conséquent perturber significativement le processus d'estimation.

Pour contourner le problème, on utilise des méthodes de mise à jour sur des séquences d'analyse partielles à la suite de (Collins, 2002). Dans ce contexte (Huang *et al.*, 2012) démontre que, dans le cas structuré, pour garantir la convergence dans le cas linéairement séparable, la mise à jour du perceptron peut se réaliser à partir de sous-séquences d'analyse qui satisfont deux conditions affaiblies. Notons $C^{(r)}_{0\Rightarrow k}$ la dérivation de référence à l'étape k et $C^{(0)}_{0\Rightarrow k} = \text{argmax}_{C_{0\Rightarrow k} \in \text{GEN}^K_k(\mathcal{T})} W(C_{0\Rightarrow k})$ la meilleure sous séquence de dérivation dans le faisceau à l'étape k. Si $C^{(0)}_{0\Rightarrow k} \neq C^{(r)}_{0\Rightarrow k}$ et que $W(C^{(0)}_{0\Rightarrow k}) > W(C^{(r)}_{0\Rightarrow k})$ alors la mise à jour :

$$\mathbf{w} \leftarrow \mathbf{w} + \mathbf{\Phi}_g(C^{(r)}_{0\Rightarrow k}) - \mathbf{\Phi}_g(C^{(0)}_{0\Rightarrow k}) \tag{7}$$

est valide. Garantir que la mise à jour est valide garantit la convergence de l'algorithme dans le cas où les données sont linéairement séparables. Dans cet article, nous examinons deux méthodes qui garantissent que la mise à jour est valide. Celles-ci diffèrent sur le choix effectif de k. La première méthode est l'*early update* (Collins, 2002) et dans ce cas, $k = \min\left(\{k | C^{(r)}_{0\Rightarrow k} \notin \text{GEN}^K_k(\mathcal{T})\}\right)$. La seconde est la *max violation update* (Huang *et al.*, 2012). En considérant l'ensemble des violations $V = \{k \mid C^{(0)}_{0\Rightarrow k} \neq C^{(r)}_{0\Rightarrow k}, W(C^{(0)}_{0\Rightarrow k}) > W(C^{(r)}_{0\Rightarrow k})\}$, on choisit $k = \text{argmax}_{k \in V}\ W(C^{(0)}_{0\Rightarrow k}) - W(C^{(r)}_{0\Rightarrow k})$

5 Le cas relâché

Dans la pratique, un certain nombre de cas d'utilisation de l'analyseur ne permettent pas d'utiliser facilement une grammaire 2-LCFG telle que supposée jusqu'ici. Il s'agit typiquement de cas où un étiqueteur morphosyntaxique est utilisé pour assigner des catégories aux mots. Dans ce type de cas, on peut souhaiter que l'analyseur utilise une séquence de tags $t_1 \dots t_n$ comme symboles terminaux. D'une part, les transformations de treebank que nous avons présentées en Section 2 peuvent potentiellement altérer ce jeu de tags (Figure 1). D'autre part, une procédure de transformation alternative qui garantit ne pas modifier le jeu de tags donne naturellement des arbres qui ne suivent pas strictement une forme normale de Chomsky (Figure 4).

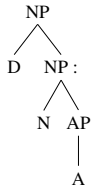


FIGURE 4 – Structure nouvelle dans le cas relâché

Certains terminaux sont introduits par des règles unaires, alors que d'autres sont introduits par des règles binaires. Par contraste avec 2-LCFG, ce type de configuration introduit naturellement de nouvelles règles de grammaire 2-CFG de la forme : $A \rightarrow B\,w, A \rightarrow w\,B$. Où w dénote un terminal pour cette grammaire (tag). Ces nouvelles formes de règles changent une propriété de 2-LCFG sur laquelle nous nous sommes appuyés jusqu'à présent : le nombre de pas de dérivation η pour dériver un arbre d'analyse d'une séquence de n mots est maintenant variable : $n-1 \leq \eta \leq 2n-1$. La conséquence est que les séquences de dérivation de l'analyseur ont une longueur η' telle que $2n-1 \leq \eta' \leq 3n-1$ et nous observons en pratique que le modèle a un biais naturel pour les séquences plus longues : celles-ci ont généralement un poids plus élevé. Pour traiter ce biais potentiel, nous formulons deux variantes de l'analyseur. La première variante, "naïve", consiste simplement à modifier la condition de terminaison de l'analyseur. Pour cela on redéfinit l'ensemble des configurations terminées $Succ = \{C_{0\Rightarrow k} | C_k = \langle n+1, \langle \sigma_e, \tau \rangle \rangle, 2n-1 \leq k \leq 3n-1\}$. Dans ce contexte, l'Equation (4) se reformule comme suit :

$$\tilde{\mathcal{C}} = \underset{C_{0\Rightarrow k} \in Succ}{\operatorname{argmax}} \; W(C_{0\Rightarrow k}) \tag{8}$$

La seconde variante de l'algorithme, dite "synchronisée", a pour but de garantir que $\eta' = 3n-1$ dans le cas pratique traité ici. Pour ce faire nous contraignons l'algorithme à réaliser nécessairement une réduction unaire ou une réduction fantôme après avoir décalé un terminal. Ce type de contrainte s'exprime en modifiant légèrement la méthode de compilation de l'automate LR décrite en Section 3. Une réduction fantôme est une nouvelle action de l'analyseur et la règle d'inférence, notée GR, associée à cette action est donnée en Figure 3. Cette règle est conçue pour synchroniser la procédure d'inférence sans changer significativement le contenu de la pile. La règle fait en quelque sorte perdre un temps à l'analyseur dans le cas où il choisit de ne pas faire de réduction unaire. Cette seconde variante ne demande pas de modifier l'équation (4).

6 Représentation structurée des mots et spécification d'un modèle d'analyse

La dernière extension que nous introduisons est motivée par la volonté de modéliser des langues à morphologie plus riche que l'anglais ou le chinois, comme par exemple le français. Pour cette famille de langues, les mots sont non seulement caractérisés par une forme mais également par un vecteur de propriétés morphologiques comme des marques d'accord en genre et en nombre, de temps, de mode, de personne, voire de cas. La richesse morphologique de ces langues contribue naturellement à augmenter le nombre de formes de mots, ce qui accroit les phénomènes de dispersion statistique. Pour cette raison, on fait l'hypothèse que l'usage d'informations extraites de dictionnaires (Mirroshandel *et al.*, 2013), de formes lemmatisées (Seddah *et al.*, 2010) ou d'aggrégats distributionnels (Candito & Crabbé, 2009) pour compléter les données d'un treebank constituent une source d'information utile à un modèle d'analyse.

Le modèle d'analyse présenté jusqu'à présent manipule les mots comme des objets atomiques encodés dans les catégories lexicalisées de la grammaire 2-LCFG. L'extension décrite ici consiste à autoriser le codage des mots par des tuples structurés de taille arbitraire. En notant $\boldsymbol{\omega}$ un tel tuple, les catégories de la grammaire 2-LCFG sont maintenant codées par des symboles de la forme $A[\boldsymbol{\omega}]$. Dans cette version étendue, les traits peuvent ainsi accéder aux différents champs de ces tuples lors de l'exécution. La taille de ces tuples est laissée libre à l'utilisateur et est en pratique fonction de la richesse des données dont il dispose en entrée du processus d'analyse. On donne à titre d'exemple indicatif en Figure 5 une représentation possible en 2-LCFG

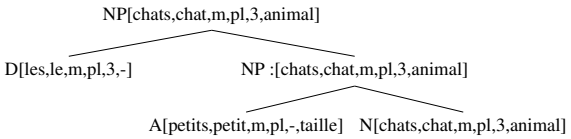


FIGURE 5 – Propagation guidée par les têtes de structures de traits lexicales

$s_0t.w_c \,\&\, s_0t.c$	$s_0t.w_f \,\&\, s_1t.w_f$	$s_0t.c \,\&\, s_1t.c \,\&\, s_2t.c$	$s_0t.c \,\&\, q_2.w_c \,\&\, q_3.w_c$	**Accord**
$s_0t.w_f \,\&\, s_0t.c$	$s_0t.w_f \,\&\, s_1t.c$	$s_0t.w_f \,\&\, s_1t.c \,\&\, s_2t.c$	$s_0t.c \,\&\, q_2.w_f \,\&\, q_3.w_c$	$s_0tc \,\&\, e(s_0t.agr, s_1t.agr) \,\&\, s_1t.c$
$s_1t.w_c \,\&\, s_1t.c$	$s_0t.c \,\&\, s_1t.w_f$	$s_0t.c \,\&\, s_1t.w_f \,\&\, q_0.w_c$	$s_0t.c \,\&\, q_2.w_c \,\&\, q_3.w_f$	$s_0tc \,\&\, e(s_0t.num, s_1t.num) \,\&\, s_1t.c$
$s_1t.w_f \,\&\, s_1t.c$	$s_0t.c \,\&\, s_1t.c$	$s_0t.c \,\&\, s_1t.c \,\&\, s_2t.w_f$	$s_0t.c \,\&\, s_0r.c \,\&\, s_1t.c$	$s_0tc \,\&\, e(s_0t.gen, s_1t.gen) \,\&\, s_1t.c$
$s_2t.w_c \,\&\, s_2t.c$	$s_0t.w_f \,\&\, q_0.w_f$	$s_0t.c \,\&\, s_1t.c \,\&\, q_0.w_c$	$s_0t.c \,\&\, s_0r.c \,\&\, s_1t.w_f$	$s_0tc \,\&\, e(s_0t.agr, q_0.agr) \,\&\, q_1.w_c$
$s_2t.w_c \,\&\, s_2t.c$	$s_0t.c \,\&\, q_0.w_f$	$s_0t.w_f \,\&\, s_1t.c \,\&\, q_0.w_c$	$s_0t.w \,\&\, s_0r.c \,\&\, s_1t.w_f$	$s_0tc \,\&\, e(s_0t.gen, q_0.gen) \,\&\, q_1.w_c$
$q_0.w_c \,\&\, q_0.w_f$	$s_0t.c \,\&\, q_0.w_c$	$s_0t.c \,\&\, s_1t.w_f \,\&\, q_0.w_c$	$s_0t.c \,\&\, s_0l.w_f \,\&\, s_1t.c$	$s_0tc \,\&\, e(s_0t.num, q_0.num) \,\&\, q_1.w_c$
$q_1.w_c \,\&\, q_1.w_f$	$q_0.w_f \,\&\, q_1.w_f$	$s_0t.c \,\&\, s_1t.c \,\&\, q_0.w_f$	$s_0t.c \,\&\, s_0l.c \,\&\, s_1t.w_f$	$s_0tc \,\&\, e(s_0t.agr, q_1.agr) \,\&\, q_1.w_c$
$q_2.w_c \,\&\, q_2.w_f$	$q_0.w_f \,\&\, q_1.w_c$	$s_0t.c \,\&\, q_0.w_c \,\&\, q_1.w_c$	$s_0t.c \,\&\, s_0l.c \,\&\, s_1t.c$	$s_0tc \,\&\, e(s_0t.num, q_0.num) \,\&\, q_1.w_c$
$q_3.w_c \,\&\, q_3.w_f$	$q_0.w_c \,\&\, q_1.w_c$	$s_0t.c \,\&\, q_0.w_f \,\&\, q_1.w_c$	**Mode**	$s_0tc \,\&\, e(s_0t.gen, q_0.gen) \,\&\, q_1.w_c$
$s_0l.w_f \,\&\, s_0l.c$	$s_1t.w_f \,\&\, q_0.w_f$	$s_0t.c \,\&\, q_0.w_c \,\&\, q_1.w_f$	$s_0t.w_m \,\&\, s_1t.w_f$	**Sous − cat**
$s_0r.w_f \,\&\, s_0r.c$	$s_1t.w_f \,\&\, q_0.w_c$	$s_0t.c \,\&\, q_1.w_c \,\&\, q_2.w_c$	$s_0t.w_f \,\&\, s_1t.w_m$	$s_0t.w_X \,\&\, s_1t.w_f$
$s_1l.w_f \,\&\, s_1l.c$	$s_1t.c \,\&\, q_0.w_f$	$s_0t.c \,\&\, q_1.w_f \,\&\, q_2.w_c$	$s_0t.c \,\&\, s_1t.w_m$	$s_0t.w_f \,\&\, s_1t.w_X$
$s_1r.w_f \,\&\, s_1r.c$	$s_1t.c \,\&\, q_0.w_c$	$s_0t.c \,\&\, q_1.w_c \,\&\, q_2.w_f$	$s_0t.w_m \,\&\, s_1t.c$	$s_0t.c \,\&\, s_1t.w_X$ $\quad s_0tw_X \,\&\, s_1t.c$

FIGURE 6 – Spécification des modèles d'analyse

relâchée d'un groupe nominal pour le français où les mots sont encodés par des tuples.

Comme il est d'usage dans la plupart des modèles de TAL, des gabarits (*templates*) sont utilisés comme langage destiné à spécifier l'ensemble des fonctions features ϕ_i à valeurs booléennes du modèle d'analyse. Nous utilisons les notations suivantes pour spécifier les gabarits du modèle d'analyse explicités en Figure (6). Les gabarits permettent d'adresser des valeurs observées relativement à une configuration de l'analyseur. Avant le point on note l'adressage d'un noeud dans une configuration. Ainsi $s_it, s_il, s_ir (0 \leq i \leq 2)$ dénotent les noeuds de la pile d'exécution avec s_0 le sommet de la pile. Un noeud de la pile s_i encode un arbre local de profondeur 1 (Figure 2) dont les différents noeuds sont adressés par les symboles t (racine), l (noeud de gauche), r (noeud de droite). q_i dénote les éléments de la file d'attente, avec q_0 le premier élément de cette file. Après le point, on note quelle valeur on extrait du noeud. c dénote la catégorie du constituant, w_f et w_c le mot tête et la catégorie du mot tête de ce constituant. On note de plus w_m, w_X le mode du mot tête et la sous-catégorie de ce mot (au sens du French Treebank), num,gen, le nombre et le genre de la tête, et agr les traits conjoints de genre et de nombre. Les gabarits élémentaires sont conjoints par le symbole $\&$ et finalement la notation $e(\cdot, \cdot)$ dénote une fonction qui renvoie vrai si les valeurs passées en argument sont égales.

7 Expériences

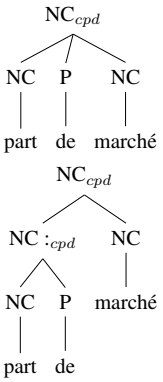
Nous présentons ici quelques expériences qui cherchent à mettrer en évidence le rôle des différents modules de l'analyseur. Celles-ci permettent également de comparer l'analyseur à l'état de l'art.

Protocole Les expériences s'appuient sur le jeux de données français SPMRL décrit dans (Abeillé *et al.*, 2003; Seddah *et al.*, 2013). Celles-ci devraient constituer le nouveau jeu de données standard pour l'analyse en constituants du français dans les années à venir et représentent un cadre plus réaliste que ceux utilisés précédemment par (Crabbé & Candito, 2008) dans la mesure où il faut également analyser les mots composés. Le jeu de données SPMRL instancie les données French Treebank dans deux scénarios : le scénario 'tags prédits' comporte un jeu de test où les tags de référence sont remplacés par des tags prédits par un tagger (exactitude d'étiquetage = 97.35%) et un scénario 'tags donnés' où le jeu de test comporte les tags de référence.

Les expériences sont menées avec une implantation de l'algorithme décrit dans l'article, écrite en C++. En particulier nous utilisons systématiquement un noyau de hachage et les mémo-fonctions décrites ci-dessus. Les données sont binarisées par une markovisation par la tête d'ordre 0, les têtes sont assignées par les heuristiques de (Arun & Keller, 2005), nous avons de plus assigné comme tête à une structure de mot composé son fils le plus à gauche. Les gabarits utilisés par défaut sont ceux spécifiés en Figure 6. Les expériences sont réalisées sur les données de développement et la comparaison avec l'analyseur de (Petrov *et al.*, 2006) est réalisée sur les données de test. Nous mesurons le F-Score et la couverture sur les données débinarisées à l'aide du logiciel `evalb` [2] et les temps reportés sont mesurés sur le jeu de test. Ils sont mesurés en secondes par phrase sur une même machine (MacOSX 2.4Ghz) pour chacun des analyseurs sans tenir compte du temps de lecture et d'écriture des données. Nous mesurons le F-Score sur les mots composés $F(\text{cpd})$ à l'aide de l'évaluateur intégré à l'analyseur de Stanford.

Chaque expérience menée fait varier une seule variable expérimentale par contraste avec une configuration de l'analyseur

2. Nous utilisons la version standard du logiciel telle que distribuée sur le site `http://nlp.cs.nyu.edu/evalb`.



Règles de dominance immédiate pour G_m^{base}		
$[a] \rightarrow [n]\,[n]$	$[n] \rightarrow [n]\,,\,[n]$	$t \rightarrow [n]\,,\,[n] \quad (\forall t \in T)$
$[a] \rightarrow [n]\,,\,[w]$	$[n] \rightarrow [n]\,,\,[w]$	$t \rightarrow [n]\,,\,[w] \quad (\forall t \in T)$
$[a] \rightarrow [w]\,[w]$	$[n] \rightarrow [w]\,,\,[w]$	$t \rightarrow [w]\,,\,[w] \quad (\forall t \in T)$
$[a] \rightarrow [w]$	$[n] \rightarrow [w]$	$t \rightarrow [n]\,,\,t \quad (\forall t \in T)$
$[a] \rightarrow [n]\,,\,t \quad (\forall t \in T)$	$[n] \rightarrow [n]\,,\,t \quad (\forall t \in T)$	$t \rightarrow [w]\,,\,t \quad (\forall t \in T)$
$[a] \rightarrow [w]\,,\,t \quad (\forall t \in T)$	$[n] \rightarrow [w]\,,\,t \quad (\forall t \in T)$	
Règles de dominance immédiate supplémentaires pour G_m^{cpd}		
$[a] \rightarrow [n]_{cpd}\,,\,[n]$	$[n] \rightarrow [n]_{cpd}\,,\,[n]$	$t \rightarrow [n]_{cpd}\,,\,[n] \quad (\forall t \in T)$
$[a] \rightarrow [n]_{cpd}\,,\,[n]_{cpd}$	$[n] \rightarrow [n]_{cpd}\,,\,[n]_{cpd}$	$t \rightarrow [n]_{cpd}\,,\,[n]_{cpd} \quad (\forall t \in T)$
$[a] \rightarrow [n]_{cpd}\,,\,[w]$	$[n] \rightarrow [n]_{cpd}\,,\,[w]$	$t \rightarrow [n]_{cpd}\,,\,[w] \quad (\forall t \in T)$
$[a] \rightarrow [n]_{cpd}\,,\,t \quad (\forall t \in T)$	$[n] \rightarrow [n]_{cpd}\,,\,t \quad (\forall t \in T)$	$t \rightarrow [n]_{cpd}\,,\,t \quad (\forall t \in T)$
$[n]_{cpd} \rightarrow [w]\,,\,[w]$	$[n]_{cpd} \rightarrow t_{cpd}\,,\,[w] \quad (\forall t_{cpd} \in T_{cpd})$	$t_{cpd} \rightarrow t_{cpd}\,,\,[w] \quad (\forall t_{cpd} \in T_{cpd})$
$[n]_{cpd} \rightarrow [w]$	$t_{cpd} \rightarrow [w]\,,\,[w] \quad (\forall t_{cpd} \in T_{cpd})$	$t_{cpd} \rightarrow [w] \quad (\forall t_{cpd} \in T_{cpd})$

FIGURE 7 – Représentation structurée des mots composés (SPMRL) et grammaires matrices correspondantes. *En haut à gauche, on a la représentation des mots composés dans le jeu de données* SPMRL. *La structure binarisée correspondante est donnée en bas à gauche. La grammaire générale $G_m^{(base)}$ encode une grammaire matrice qui ne tient pas spécialement compte des mots composés et dont les classes d'équivalence sont $[a]$ le symbole axiome, $[n]$ les symboles non terminaux non temporaires et $[w]$ l'ensemble des terminaux. Chaque non terminal temporaire $t \in T$ forme sa propre classe d'équivalence. La grammaire $G_m^{(cpd)}$ distingue en plus une classe d'équivalence $[n]_{cpd}$, représentant les non terminaux marqués comme composés, disjointe de la classe $[n]$ des terminaux non marqués comme tels. $T_{(cpd)}$ est un ensemble de symboles temporaires marqués comme composés qui est disjoint de T.*

par défaut. La configuration par défaut pose que la taille du faisceau $K = 4$, que la gestion du relâchement utilise la version naïve (Section 5), que la grammaire utilise la construction LR $G_m^{(base)}$ (Section 3 et Figure 7), que la mise à jour est l'*early update* (Section 4) et que l'intégralité des gabarits donnée en Figure 6 est utilisée. Le protocole spécifique à chaque expérience est le suivant. *(1) Taille du Beam* La première expérience fait varier la taille du faisceau. Nous testons différentes valeurs de la constante K qui fixe la taille du faisceau ($\text{GEN}_k^K(\mathcal{T})$ ci-dessus) pour $K = 2, K = 4$ (défaut), $K = 8, K = 16$. *(2) Relâchement* Par défaut, nous utilisons l'analyseur en mode naïf comme spécifié en Section 5 dans les différentes expériences. Pour l'expérience de relâchement, nous testons le mode naïf (*naive*) et le mode synchronisé (*sync*) comme décrit en Section 5. *(3) Grammaire* Nous faisons varier le type de grammaire matrice utilisée (Section 3) pour générer l'automate LR en utilisant les grammaires matrices présentées en (Figure 7). Nous contrastons en particulier une grammaire générale $G_m^{(base)}$ avec une grammaire $G_m^{(cpd)}$ qui comporte une sous-grammaire spécifique pour le traitement des mots composés. *(4) Mise à jour* Cette expérience fait varier la méthode de mise à jour comme décrit en Section 4. On teste l'influence de la mise à jour rapide (*early update*) par contraste avec la mise à jour à violation maximale (*max violation*). Les modèles entrainés avec la mise à jour rapide le sont sur 25 époques. Les modèles avec mise à jour à violation maximale sont entraînés sur 12 époques. *(5) Morphologie* Cette expérience teste l'impact des gabarits morphologiques. On contraste le modèle qui comporte l'ensemble des gabarits morphologiques (Figure 6), modélisant notamment l'accord avec un modèle plus pauvre ou les gabarits rangés sous les sections *accord, sous-cat, mode* en Figure 6 sont ignorés. *(6) Analyseur de Berkeley* Il s'agit de comparer l'analyseur décrit dans cet article avec l'analyseur de Berkeley (Petrov *et al.*, 2006) connu pour représenter l'état de l'art en termes de rapidité et de correction des analyses. Nous réutilisons ici les résultats donnés par (Seddah *et al.*, 2013) sur le jeu de test avec cet analyseur en utilisant le score `evalb` standard.

Résultats et discussions Très généralement, les expériences contribuent à justifier la méthode d'analyse proposée dans cet article. Celle-ci se fonde sur un algorithme à décalage réduction et une méthode d'inférence approximative en faisceau pour l'analyse en constituants. Les résultats obtenus sont état de l'art en temps comme en exactitude, en excluant divers résultats obtenus par mélange d'analyseurs et par utilisation de ressources exogènes. Si en ce qui concerne l'exactitude, les différences avec l'analyseur de Berkeley (Petrov *et al.*, 2006) sont faibles (Table 3), on constate que l'apport principal de la méthode décrite ici est son efficacité en temps (Figure 8).

On constate plus spécifiquement que c'est la prise en compte des informations morphologiques qui permet à l'analyseur d'être état de l'art $F_{\leq 40} = 87.02$ pour $K = 4$ et $F_{\leq 40} = 87.14$ pour $K = 8$ avec des faisceaux de petite taille et des temps d'analyse très faibles ($t_\mu = 0.06s, t_{\text{max}} = 0.3s$ pour $K = 4$. et $t_\mu = 0.1s, t_{\text{max}} = 0.5s$ pour $K = 8$) par comparaison avec l'existant : Berkeley (Petrov *et al.*, 2006) $t_\mu = 0.28s, t_{\text{max}} = 10.27s$. Ce dernier est reporté par (Huang & Sagae, 2010) comme plus rapide que (Charniak, 2000). Autrement dit l'expressivité ajoutée par le modèle permet de compenser l'approximation introduite pour le rendre efficace.

Dev(tags donnés)			
Expérience	F≤ 40	F	Cov
K=2	85.40	82.74	98.6
K=4	86.52	83.69	99.5
K=8	86.80	84.31	99.9
K=16	86.49	83.95	99.9
no-morph	85.23	82.43	99.8
all-morph	86.52	83.69	99.5
sync	86.41	83.66	99.6
naive	86.52	83.69	99.5
cpd	86.30	83.30	99.2
base	86.52	83.69	99.5
Max Violation	85.98	83.49	99.5
Early Update	86.52	83.69	99.5

Dev(tags prédits)			
Expérience	F≤ 40	F	Cov
K=2	83.24	80.42	98.9
K=4	84.32	81.34	99.4
K=8	84.43	81.79	99.8
K=16	84.59	81.94	99.8
no-morph	83.68	81.05	99.8
all-morph	84.32	81.34	99.4
sync	84.06	81.14	99.9
naive	84.32	81.34	99.4
cpd	83.84	81.17	99.0
base	84.32	81.34	99.4
Max Violation	83.42	80.56	99.5
Early Update	84.32	81.34	99.4

Test (tags donnés)	F≤ 40	F	Cov	F(cpd)
K=4	87.02	83.99	99.7	77.48
K=8	87.14	84.20	99.8	77.26
Berkeley	86.44	83.96	99.9	73.38
Test (tags prédits)	F≤ 40	F	Cov	F(cpd)
K=4	84.03	80.98	99.7	69.39
K=8	84.33	81.43	99.8	70.26
Berkeley	83.16	80.73	99.9	69.32
Test (texte brut)	F≤ 40	F	Cov	F(cpd)
Berkeley	83.59	81.33	99.9	70.25

TABLE 3 – Résultats expérimentaux

Il est par contre plus surprenant de constater le résultat nul concernant la synchronisation de l'analyseur. La version naïve se comporte même un peu mieux que la version explicitement synchronisée. Il est possible que le jeu de gabarits qui a été mis au point principalement sur le modèle d'exécution naïf procure un avantage à ce dernier.

On remarque que le modèle comportant une grammaire locale spécifique aux mots composés $G_m^{(cpd)}$ a une couverture plus faible que la grammaire $G_m^{(base)}$. Par contre on remarque qu'il est significativement plus rapide ($t_\mu = 0.04s, t_{\max} = 0.2s$) . La grammaire est plus contrainte et l'automate comporte moins d'états de succès. On pense travailler à l'avenir pour définir une méthode de recherche de solutions garantissant la complétude de l'algorithme d'analyse pour permettre de combiner la grammaire générale avec des grammaires locales pour mots composés (ou entités nommées par exemple).

En ce qui concerne, la méthode de mise à jour, l'observation principale concerne la courbe d'apprentissage. L'algorithme converge beaucoup plus vite pour la méthode *Max Violation* (entre 10 et 15 itérations contre 25-30 pour *Early update*). Par contre la méthode *Max Violation* a une tendance au surentrainement encore plus prononcée que *Early Update*, ce qu'il faut améliorer à l'avenir.

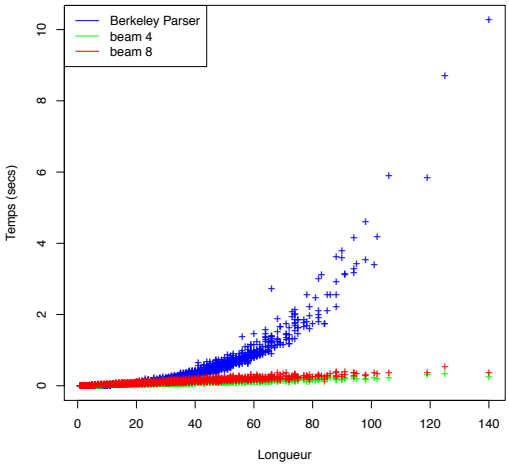


FIGURE 8 – Temps d'analyse

8 Conclusion

À notre connaissance, l'article propose le premier algorithme d'analyse syntaxique déterministe en constituants pour le langage naturel guidé par un automate LR et pondéré par un modèle discriminant. L'article montre que l'usage d'un modèle d'analyse très expressif, qui permet notamment de capturer des informations morphologiques, est rendu très efficace par l'usage d'une approximation heuristique. Celle-ci permet d'obtenir une exécution en temps linéaire tout en obtenant des résultats état de l'art en exactitude pour l'analyse syntaxique du français. La suite des travaux va porter principalement sur l'intégration d'une construction sémantique conjointe à l'analyse syntaxique et sur l'intégration de ressources exogènes dans le modèle d'analyse dans le but de créer un analyseur sémantique capable d'analyser efficacement de gros volumes de données textuelles issues du web.

Remerciements

L'auteur remercie Maximin Coavoux qui a contribué au développement, Benoît Sagot pour ses encouragements et les diverses discussions ayant permis de clarifier le propos, et finalement Djamé Seddah pour son aide sur les données.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building a treebank for French. In *Treebanks*. Kluwer.

AHO A. V., SETHI R., ULLMAN J. D. & LAM M. S. (2006). *Compilers : Principles, Techniques, and Tools*. Addison Wesley.

ARUN A. & KELLER F. (2005). Lexicalization in crosslinguistic probabilistic parsing : The case of French. In *Association for Computational Linguistics*.

CANDITO M. & CRABBÉ B. (2009). Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of International Workshop on Parsing Technologies (IWPT)*.

CHARNIAK E. (2000). A maximum-entropy-inspired parser. In *North American Association for Computational Linguistics*.

CHARNIAK E. & JOHNSON M. (2005). Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*.

COLLINS M. (2002). Discriminative training methods for hidden markov models : Theory and experiments with perceptron algorithms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

COLLINS M. (2003). Head-driven statistical models for natural language parsing. *Computational Linguistics*, **29**(4).

CRABBÉ B. & CANDITO M. (2008). Expériences d'analyses syntaxique statistique du français. In *Actes deTALN 2008*.

FINKEL J. R., KLEEMAN A. & MANNING C. D. (2008). Efficient, feature-based, conditional random field parsing. In *Proceedings of the Association for Computational Linguistics*.

GAZDAR G., KLEIN E., PULLUM G. K. & SAG I. A. (1985). *Generalized Phrase Structure Grammar*. Cambridge, MA : Harvard University Press and Oxford : Basil Blackwell's.

HUANG L., FAYONG S. & GUO Y. (2012). Structured perceptron with inexact search. In *North American Association for Computational Linguistics*.

HUANG L. & SAGAE K. (2010). Dynamic programming for linear-time incremental parsing. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*.

KNUTH D. (1965). On the translation of languages from left to right. *Information and Control*, **8**(6).

MIRROSHANDEL S. A., NASR A. & SAGOT B. (2013). Enforcing subcategorization constraints in a parser using subparses recombining. In *Conference of the North American Chapter of the Association for Computational Linguistics*.

NEDERHOF M.-J. & SATTA G. (2010). Algorithmic aspects of natural language processing. In M. ATALLAH & M. BLANTON, Eds., *Algorithms and Theory of Computation Handbook*. CRC press.

PETROV S., BARRETT L., THIBAUX R. & KLEIN D. (2006). Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia : Association for Computational Linguistics.

SAGAE K. & LAVIE A. (2006). A best-first probabilistic shift-reduce parser. In *Proceedings of the COLING/ACL*.

SEDDAH D., CHRUPALA G., CETINOGLU O., VAN GENABITH J. & CANDITO M. (2010). Lemmatization and lexicalized statistical parsing of morphologically rich languages : the case of French. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically Rich Languages*.

SEDDAH D., TSARFATY R., KÜBLER S., CANDITO M., CHOI J. D., FARKAS R., FOSTER J., GOENAGA I., GOJENOLA GALLETEBEITIA K., GOLDBERG Y., GREEN S., HABASH N., KUHLMANN M., MAIER W., NIVRE J., PRZEPIÓRKOWSKI A., ROTH R., SEEKER W., VERSLEY Y., VINCZE V., WOLIŃSK M., WRÓBLEWSKA A. & VILLEMONTE DE LA CLERGERIE É. (2013). Overview of the SPMRL 2013 Shared Task : A Cross-Framework Evaluation of Parsing Morphologically Rich Languages. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*.

SHI Q., DROR G., LANGFORD J., SMOLA A. & VISHWANATHAN S. (2009). Hash kernels for structured data. *Journal of Machine Learning Research*, **10**.

SOCHER R., HUVAL B., MANNING C. D. & NG A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Conference on Empirical Methods in Natural Language Processing*.

TOMITA M. (1985). An efficient context free parsing algorithm for natural language. In *International Joint Conference on Artificial Intelligence (IJCAI)*.

TOMITA M. (1988). Graph structured stack and natural language parsing. In *Proceedings of the Association for Computational Linguistics (ACL)*.

ZHANG Y. (2009). Transition-based parsing of the chinese treebank using a global discriminative model. In *Proceedings International Workshop on Parsing Technologies*.

ZHU M., ZHANG Y., CHEN W., ZHANG M. & ZHU J. (2013). Fast and accurate shift-reduce constituent parsing. In *Association for Computational Linguistics*.

Détection automatique de la structure organisationnelle de documents à partir de marqueurs visuels et lexicaux

Jean-Philippe Fauconnier[1] Laurent Sorin[1] Mouna Kamel[1]
Mustapha Mojahid[1] Nathalie Aussenac-Gilles[1]
(1) IRIT, Université Paul Sabatier, 118 Route de Narbonne, 31062 Toulouse Cedex 9
{prénom}.{nom}@irit.fr

Résumé. La compréhension d'un texte s'opère à travers les niveaux d'information visuelle, logique et discursive, et leurs relations d'interdépendance. La majorité des travaux ayant étudié ces relations a été menée dans le cadre de la génération de textes, où les propriétés visuelles sont inférées à partir des éléments logiques et discursifs. Les travaux présentés ici adoptent une démarche inverse en proposant de générer automatiquement la structure organisationnelle du texte (structure logique) à partir de sa forme visuelle. Le principe consiste à (i) labelliser des blocs visuels par apprentissage afin d'obtenir des unités logiques et (ii) relier ces unités par des relations de coordination ou de subordination pour construire un arbre. Pour ces deux tâches, des *Champs Aléatoires Conditionnels* et un *Maximum d'Entropie* sont respectivement utilisés. Après apprentissage, les résultats aboutissent à une exactitude de 80,46% pour la labellisation et 97,23% pour la construction de l'arbre.

Abstract. The process of understanding a document uses both visual, logic and discursive information along with the mutual relationships between those levels. Most approaches studying those relationships were conducted in the frame of text generation, where the text visual properties are infered from logical and discursive elements. We chose in our work to take the opposite path by trying to infer the logical structure of texts using their visual forms. To do so, we (i) assign a logical label to each visual block and (ii) we try to connect those logical units with coordination or subordination relationships, in order to build a logical tree. We used respectively a *Conditional Random Fields* and a *Maximum Entropy* algorithms for those two tasks. After a learning phase, the obtained models give us a 80,46% accuracy for task (i) and a 97,23% accuracy for task (ii).

Mots-clés : discours, structure organisationnelle, mise en forme matérielle, marqueurs métadiscursifs, champs aléatoires conditionnels, maximum d'entropie.

Keywords: discourse, organizational structure, text formating, metadiscursive markers, conditional random fields, maximum entropy.

1 Introduction

La construction automatique de la structure de documents constitue un enjeu majeur en Traitement Automatique du Langage (TAL). En effet, les traitements des modules actuels (e.g : étiquetage morpho-syntaxique, reconnaissance d'entités nommées, etc.) opèrent généralement à un niveau de granularité qui ne prend pas en compte les phénomènes se déroulant au niveau supérieur, tels que les relations entre les sections, titres, paragraphes, etc. (Marcu, 2006). Or, une approche plus globale des textes paraît être une étape nécessaire pour améliorer l'accessibilité des documents (Sorin *et al.*, 2013), la navigation intra-documentaire (Couto *et al.*, 2004), le résumé automatique (Bossard, 2009) ainsi que l'extraction d'information (Fauconnier *et al.*, 2013).

Nous partons du constat qu'un texte peut être segmenté selon trois structures : (i) *la structure visuelle* (segmentation en pages, blocs visuels, etc.), (ii) *la structure logique* (segmentation en titres, paragraphes, etc.), et (iii) *la structure discursive* (segmentation en unités élémentaires et complexes du discours). Les frontières entre ces structures ne sont pas nettement établies dans la littérature. Toutefois, il est admis que ces structures s'échelonnent graduellement dans la compréhension d'un texte et entretiennent des relations complexes d'interdépendance. Par exemple, la mise en forme spatiale d'un texte a des répercussions sur l'interprétation de sa structure logique (Virbel *et al.*, 2005), et une relation logique de coordination entre deux items d'une structure hiérarchique implique une relation rhétorique spécifique (Vergez-Couret *et al.*, 2011).

L'analyse des structures de documents est un sujet traité au sein de la communauté de l'*Analyse de Documents* (conférences ICDAR, IJDAR, etc.). Généralement, cette tâche est vue comme un problème d'analyse syntaxique et un arbre ordonnant les unités du document est attendu en sortie (Mao *et al.*, 2003). Deux domaines sont considérés : l'analyse géométrique (Tokuyasu & Choub, 2001) et l'analyse logique (Klink *et al.*, 2000). Toutefois, les représentations logiques obtenues ne sont souvent pas adaptées à une analyse fine au niveau discursif. Cette difficulté apparaît lorsque des objets textuels complexes conjuguent à la fois mise en forme matérielle et phénomènes discursifs (e.g : structures hiérarchiques imbriquées, définitions, etc.). En outre, les labels logiques ne sont pas toujours fins (Aiello *et al.*, 2002) et il n'existe pas de consensus sur les valeurs qu'ils peuvent prendre. Cela s'explique notamment par un intérêt portant davantage sur l'analyse géométrique (Paaß & Konya, 2012), plus compliquée pour les documents historiques, les lettres, etc., et non sur la construction d'une structure logique en lien avec la structure discursive.

Au sein de la communauté TAL, les dernières années ont montré un intérêt pour les documents (Péry-Woodley & Scott, 2006) et plusieurs approches pour la structuration de ceux-ci en lien avec le discours ont été proposées. Citons la *Document Structure* (Power *et al.*, 2003), le système *DArt*$_{bio}$ (Bateman *et al.*, 2001) et le *Modèle d'Architecture Textuelle* (Luc & Virbel, 2001). Ces trois approches reposent sur la *Rhetorical Structure Theory* (RST) (Mann & Thompson, 1988). Cependant, bien que ces approches offrent des cadres théoriques poussés, elles ont pour vocation dans leurs implémentations actuelles la génération automatique de textes. L'élaboration de la structure visuelle est généralement faite au travers d'une correspondance entre les structures logiques et discursives données en entrée. À notre connaissance, il n'existe pas d'implémentation opérant le procédé inverse dans une optique discursive.
Notons que d'autres recherches ont visé à produire une structuration des textes dans l'étude de phénomènes locaux, telles que les ruptures thématiques (Choi, 2002; Couto *et al.*, 2004) ou encore les structures fines de texte (Hernandez & Grau, 2005). Cependant, ces approches ne traitent pas la structuration hiérarchique du document dans sa globalité. À l'inverse, (Ratté *et al.*, 2007) proposent un système symbolique pour l'analyse de documents, mais se limitent aux titres, aux chapitres et aux énumérations de premier niveau (non imbriquées) sans proposer de liens entre ces éléments.

Dans ce travail, nous proposons une représentation en arbre du document au travers de relations métadiscursives, appelée structure organisationnelle. Ces relations sont dites métadiscursives car elles ne dépendent pas du contenu propositionnel des unités logiques qu'elles lient (e.g : titres, paragraphes, items, citations, etc.). Nous représentons ces relations par deux relations de dépendance : la subordination et la coordination. L'avantage premier de cette représentation réside dans la détermination du rôle joué par les éléments logiques dans l'ensemble du document. Par exemple, un élément labellisé comme paragraphe peut avoir un rôle d'item dans une structure plus large. Ceci ouvre notamment la voie à une tâche ultérieure visant la reconnaissance de phénomènes complexes agençant plusieurs unités (e.g : structures hiérarchiques à imbrications multiples, etc.). Pour construire cet arbre, notre méthode prend en entrée des documents PDF préalablement traités par une analyse géométrique avec l'outil LA-PDFText (Ramakrishnan *et al.*, 2012) et procède en deux étapes : (i) la reconnaissance des unités logiques au sein des documents et (ii) la construction de l'arbre liant ces unités logiques.

Dans la section 2, nous décrivons les différentes structures et les situons dans les approches existantes. L'arbre en dépendance est décrit en section 3. Nous présentons le corpus en section 4. Les étapes de traitement sont décrites en section 5 et évaluées en section 6. Une discussion est proposée en section 7. Enfin, nous concluons ce travail sur quelques perspectives.

2 Définitions et modèles pour la structuration de documents

Bien que la majorité des travaux s'accorde sur le fait que plusieurs niveaux de structuration existent (visuel, logique et discursif), il n'existe pas de véritable consensus quant aux frontières entre ces niveaux. Nous proposons de définir ces structures et ajoutons la notion de **structure logique profonde** qui correspond à notre structure organisationnelle (Section 2.1). Ensuite, nous montrons dans quelle mesure celle-ci s'intègre dans l'un des modèles préexistants en structuration de documents liés au discours (Section 2.2).

2.1 Définitions des structures

Nous définissons la **structure visuelle** d'un document comme la forme visuelle dans laquelle il apparaît. Les unités visuelles sont identifiées par des indices de nature typographique ou dispositionnelle qui peuvent suivre une convention liée au support, au moyen de production ou au mode divulgation du document. L'*unité élémentaire* est l'alinéa, c'est-à-dire un segment textuel encadré par deux moyens dispositionnels (e.g : retours à la ligne, etc.). Plusieurs alinéas peuvent composer un bloc visuel, dit aussi *unité visuelle complexe*, lorsque l'écart les séparant est plus petit ou égal à l'interligne.

La **structure logique** d'un document se définit comme un niveau abstrait ordonnant le document en *unités logiques élémentaires* et *unités logiques complexes*. Ces unités sont dites logiques, car elles participent à la compréhension du texte en y jouant un rôle métadiscursif, c'est-à-dire indépendant de leur contenu propositionnel. À ce niveau, nous posons pour les besoins de l'analyse deux sous-structures dont la distinction est graduelle :

— La **structure logique de surface** d'un document est composée d'*unités logiques élémentaires*. Ces unités peuvent être un titre, un paragraphe, une note de bas de page, une citation, une référence bibliographique, mais aussi l'alinéa. À ce niveau, le nom de chacune de ces unités dénote le rôle métadiscursif (ou son absence pour l'alinéa) qu'elle joue dans le texte. Cette liste correspond en partie à ce qui est proposé dans les langages de balisage tels que HTML ou LaTeX, où une distinction est faite entre contenu et mise en forme. Pour des raisons pratiques, ces langages permettent de représenter l'alinéa sans pour autant qu'il soit balisé. Notons que (Power *et al.*, 2003) proposent une description des liens entre la *structure logique de surface* et les langages de balisage.

— La **structure logique profonde** correspond à la **structure organisationnelle** d'un document. Celle-ci ordonne les unités logiques élémentaires en *unités logiques complexes* et correspond à l'organisation du document telle que voulue par son auteur. Les unités complexes peuvent être des sections, des structures hiérarchiques, etc., et peuvent s'imbriquer, se chevaucher ou encore se superposer. Au sein de cette structure, un phénomène de changement de rôle peut apparaître. Une unité considérée comme paragraphe lorsqu'elle est prise isolément peut endosser le rôle d'item au sein d'une structure hiérarchique. Ceci survient lorsque ce paragraphe n'est pas mis en forme comme un item et présente des connecteurs tels que « Premièrement », « Deuxièmement », etc. À ce niveau, les unités entretiennent entre elles des relations complexes que nous qualifions de métadiscursives.

Enfin, la **structure discursive** d'un document est la structure qui ordonne son *message*. Les *unités élémentaires* et *complexes* de discours sont liées les unes aux autres par des relations rhétoriques (Mann & Thompson, 1988; Asher, 1993). Il existe une interdépendance forte entre les unités de discours et les unités logiques, car le contexte d'apparition d'une unité influence le rôle qu'elle joue dans la compréhension d'un texte.

2.2 Modèles pour la structuration de documents

Bien qu'initialement orientées pour la génération de textes, les trois approches présentées ci-dessous proposent un cadre théorique utile pour l'analyse de la structuration de documents :

La *Document Structure* de (Power *et al.*, 2003) est définie comme un niveau abstrait et séparé dans la description d'un document. Ce niveau logique se positionne entre la *représentation physique*, qui correspond à la structure visuelle, et le *message*, qui correspond à la structure discursive. Cette théorie voit son origine dans la *Text-grammar* de (Nunberg, 1990) qui différencie la *phrase syntaxique* (contrainte par la grammaire syntaxique) de la *phrase textuelle* (chaîne de caractères commençant par une majuscule et se terminant par un point). La *Document Structure* étend cette distinction et propose plusieurs *unités abstraites* dont l'unité élémentaire est équivalente à l'alinéa. Ces unités sont hiérarchisées selon des critères de composition et forment pour chaque document un arbre en constituants.

Le système de génération automatique de biographies DArt_{bio} proposé par (Bateman *et al.*, 2001) repose sur un modèle distinguant aussi représentation physique (*page layout*), structure logique (*layout structure*) et structure discursive (*rhetorical structure*). La *layout structure* diffère de la *Document Structure* principalement en deux points : (i) le bloc est le seul type d'unité considéré et (ii) l'ordonnancement des blocs ne repose pas sur des critères de composition. Le cadre théorique est riche (e.g : mise en forme avec images, tables, etc.), mais difficilement utilisable hors du contexte des biographies.

Le *Modèle d'Architecture Textuelle* (MAT) (Luc & Virbel, 2001) a pour vocation de représenter les phénomènes architecturaux des textes au travers d'un métalangage Harrissien (Harris, 1971). Ce métalangage permet d'organiser des *objets textuels*, définis comme des segments rendus perceptibles à la surface du texte. Par exemple, la métaphrase suivante :

L'auteur intitule texte(1) par un titre identifié en titre(1)

indique que l'*objet textuel* identifiée comme *titre(1)* endosse le rôle métadiscursif de titre pour *texte(1)*. L'ensemble des métaphrases forment un *graphe architectural* et correspond à la structure logique (de surface et profonde). Le choix des objets textuels et des relations se fait avec la *mise en forme matérielle* qui regroupe les propriétés de réalisation d'un texte.

Nous pensons que le MAT est le modèle le plus apte à représenter la structure organisationnelle. Bien que ce modèle soit nativement orienté vers la génération de textes, il reste un modèle ouvert permettant la description d'*unités logiques complexes* hiérarchiques et transversales. De plus, contrairement à (Power *et al.*, 2003) et (Bateman *et al.*, 2001), une correspondance est faite entre les marqueurs de *mise en forme matérielle* et l'*architecture textuelle* (structure logique).

Ces marqueurs, dits métadiscursifs, se réalisent sous trois formes : (i) les marqueurs dispositionnels tels que les retours à la ligne, les retraits, etc., (ii) les marqueurs typographiques tels que les puces, les numérotations, etc., et (iii) les marqueurs lexicaux correspondant notamment aux marqueurs d'intégration linéaire (MIL) (e.g :« Premièrement », « Deuxièmement », etc.). Ces trois formes peuvent être combinées dans la réalisation d'une même unité logique. Notons qu'une équivalence existe entre la dernière famille et les *introducteurs de cadres* de (Charolles, 1997). Dans la suite, nous utilisons le MAT comme cadre théorique et employons les marqueurs décrits dans ce modèle pour l'analyse de documents.

3 Représentation hiérarchique de la structure organisationnelle

Nous représentons la structure organisationnelle par un arbre en dépendance organisant hiérarchiquement les *unités logiques élémentaires*, telles que les titres, les paragraphes, les items, etc. (Section 2.1). De manière comparable aux travaux de (Choi, 2002) et (Hernandez & Grau, 2005) sur les énoncés, nous proposons de représenter les relations entre unités par des relations de *subordination* et de *coordination*. Une même relation de subordination est partagée par deux unités coordonnées. Et nous posons le nœud factice *texte* comme racine de l'arbre.
Le principe de dépendance suivi consiste à articuler ensemble les unités qui apparaissent liées dans la cohésion du document. Le choix d'une relation entre deux unités se fait sur la base du changement de niveau dans le texte et, parallèlement, par la présence des marqueurs métadiscursifs qui le marquent. Une relation de subordination désigne une descente dans la structure du document, et une relation de coordination lie deux unités partageant le même niveau et le même label.

Cette représentation a l'avantage d'être indépendante de l'ordre *a priori* entre les labels logiques des nœuds. Un élément considéré comme paragraphe peut être subordonné à un item s'il comporte des indices lexicaux qui indiquent cette dépendance. Également, cette représentation ouvre la voie à l'identification ultérieure de phénomènes complexes (e.g : structures hiérarchiques à niveaux multiples, etc.) par parcours d'arbre. Enfin, l'intégration au graphe architectural du MAT sous la forme d'un sous-arbre permettra de typer finement les phénomènes extraits. Toutefois, notons que la représentation proposée dans ce travail est un modèle simplifié et la sémantique réelle des relations n'est pas traitée ici. Citons néanmoins les travaux de (Bouayad-Agha *et al.*, 2000) et (Lüngen *et al.*, 2010) qui abordent cette problématique.

Dans la figure 1, nous proposons deux exemples de correspondance entre un document et sa structure organisationnelle. Dans chacun d'eux, le document est représenté par un schéma où les paragraphes débutent par une majuscule, les items par une puce et les titres sont numérotés. Dans l'arbre, les relations de subordination sont représentées par des arcs continus et les relations de coordination par des arcs en pointillé.

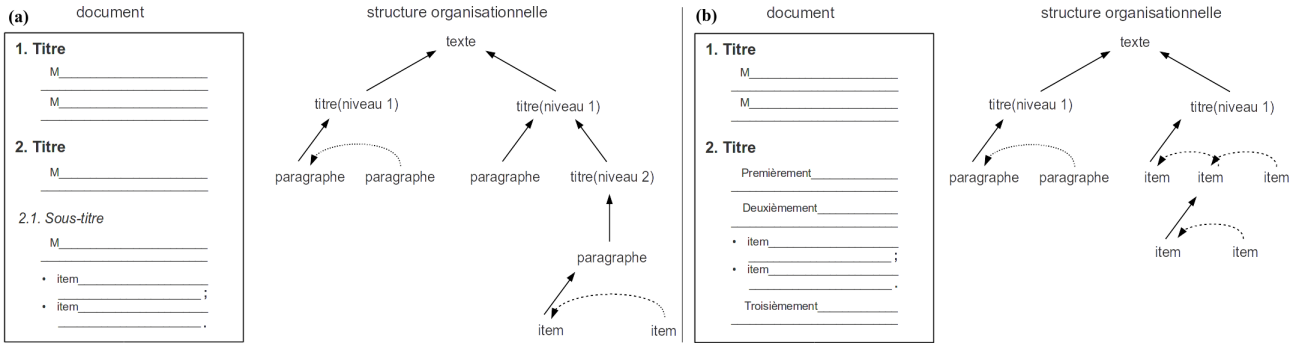


FIGURE 1 – Correspondances entre documents et structures organisationnelles

L'exemple (a) présente une organisation hiérarchique prenant en compte la titraille, ainsi qu'une structure multi-échelle. Une structure multi-échelle, définie par (Ho-Dac *et al.*, 2010), est une *unité logique complexe* qui peut apparaître à tous les niveaux du documents (intra-paragraphique [1], multi-paragraphique, sous-section, section, etc.). Elle se compose d'une amorce, d'une énumération comprenant au-moins deux items, et optionnellement d'une clôture. En organisant leurs items au sein d'une relation d'égalité selon un critère de *coénumérabilité* (implicite ou explicite), les structures multi-échelles participent à la *cohésion textuelle* (Péry-Woodley *et al.*, 2011). Dans (a), la structure multi-échelle est mise en forme visuellement par des indices dispositionnels (les retraits) et des indices typographiques (les puces). L'exemple (b) montre deux structures multi-échelles imbriquées. La première présente des items qui localement endossent le rôle de paragraphe et qui sont introduits par des marqueurs lexicaux. La seconde, qui est imbriquée, est mise en forme comme dans (a).

1. Les structures multi-échelles intra-paragraphiques ne sont pas traitées dans le cadre de ce travail.

4 Construction d'un corpus enrichi visuellement et logiquement

Dans l'objectif d'implémenter des approches par apprentissage supervisé, il a d'abord été nécessaire de construire un corpus riche en marqueurs visuels (typographiques et dispositionnels) et lexicaux. Les corpus LING et GEOP, inclus dans le corpus ANNODIS (Péry-Woodley *et al.*, 2011), ont été choisis comme point de départ car ils présentent deux propriétés : (i) une représentation native au format PDF et (ii) une annotation des structures multi-échelles.

Le corpus LING est constitué de 25 articles scientifiques issus des actes du CMLF 2008 [2]. Le corpus GEOP est constitué de 21 [3] rapports/articles de l'IFRI [4]. Ces deux corpus permettent d'expérimenter deux terrains spécifiques. Au niveau de la structure visuelle, LING présente une mise en forme unifiée (convention du CMLF), alors que GEOP présente des documents très hétérogènes. Au niveau de la structure organisationnelle, LING est relativement complexe, présentant notamment de nombreuses structures multi-échelles imbriquées, tandis que GEOP est plus linéaire.

Nous avons enrichi semi-automatiquement ces corpus par des annotations relatives à (1) leur structure visuelle, (2) leur structure logique de surface et, enfin, (3) leur structure logique profonde. Ce travail a été réalisé en trois étapes successives :

(1) Les documents au format PDF ont été segmentés en blocs visuels en utilisant la segmentation automatique proposée par l'outil LA-PDFText (Ramakrishnan *et al.*, 2012). Cette analyse géométrique repose sur un algorithme qui calcule la proximité entre blocs de mots en prenant en compte leur position mais aussi leurs caractéristiques typographiques locales (fonte, police). Une fois qu'un seuil calculé automatiquement pour chaque page est dépassé, deux blocs de mots sont agrégés. Selon ce principe, cet algorithme segmente de manière ascendante chaque page en une série de blocs visuels.
Toutefois, les blocs proposés par cet outil présentant de nombreuses erreurs (e.g : des paragraphes coupés en deux, inversions dans les blocs, notes de bas de page agglomérées, etc.), un traitement manuel de l'ensemble du corpus a été effectué.
Au terme de cette étape, chacun des blocs visuels contenus dans les documents est caractérisé dispositionnellement et, lorsqu'il s'agit d'un mot, typographiquement. La figure 2 présente un extrait du corpus, où les attributs (x_1, y_1) et (x_2, y_2) représentent respectivement les coordonnées (en pixels) du coin supérieur gauche d'un bloc et de son coin inférieur droit.

```
<page x1="70" y1="71" x2="524" y2="806">
  <chunk x1="70" y1="346" x2="524" y2="360">
    <word x1="106" y1="346"... font="Arial" style="16pt;Bold">Le</Word>
    <word x1="135" y1="346"... font="Arial" style="16pt;Bold">sens</Word>
      ...
  </chunk>
</page>
```

FIGURE 2 – Exemple XML des propriétés visuelles d'un document

(2) Chaque bloc visuel a été annoté avec un label logique élémentaire. Les labels choisis ici sont les titres (de niveau 1 à 3), les paragraphes, les items, les citations, les en-têtes et les pieds de page, les bylines [5], les notes de bas de page et, enfin, les références bibliographiques. Un label *autres* a été choisi pour classer par défaut les blocs non textuels (e.g : images, tables, etc.). Cette étape de classification des blocs visuels comprend deux temps. Premièrement, une annotation a été réalisée avec l'algorithme de similarité textuelle décrit dans (Myers, 1986) en associant les labels logiques du corpus ANNODIS originel aux blocs visuels de notre corpus. Deuxièmement, à l'aide d'une interface en ligne de commande, les labels non traités dans ANNODIS (e.g : en-têtes, notes de bas de page, etc.) ont été ajoutés manuellement.
Au terme de cet enrichissement, des différences significatives (calculées par un χ^2 avec un α à 0,001) apparaissent dans les distributions de labels de LING et GEOP (Tableau 1). Le caractère linguistique de LING implique un plus grand nombre d'items (dont 210 dédiés à l'énumération d'exemples linguistiques tirés de corpus). Son caractère académique implique aussi un grand nombre de citations et de références bibliographiques. Le caractère visuellement hétérogène de GEOP s'observe au travers du grand nombre d'en-têtes et pieds de page, ainsi que dans la présence de nombreuses unités appartenant à la classe *autres* (images et diagrammes). De manière transversale, le paragraphe est l'unité la plus représentée, légèrement en plus grand nombre dans GEOP dont les articles se veulent plus littéraires.

(3) Les deux corpus ont été enrichis par la structure organisationnelle des documents. Cet enrichissement a été effectué en deux temps. Premièrement, des arbres hiérarchiques en dépendance ont été générés à partir d'une grammaire formelle décrivant les relations *a priori* entre les unités logiques élémentaires. Par exemple, un item est subordonné au paragraphe

2. Congrès Mondial de Linguistique Française
3. Sur les 32 articles de GEOP, 21 ont été sélectionnés car considérés comme représentatifs des propriétés visuelles et logiques du corpus.
4. Institut Français de Relations Internationales
5. Le terme byline est un terme générique utilisé pour désigner les lignes de texte en début de document consacrées à l'auteur, sa position et la date.

	h(1,2,3)	para.	item	cit.	en-tête	pied p.	byline	note p.	bibl	autres	Total
LING	304	1241	380	123	45	16	80	394	1173	82	3838
Moy.	12,1	49,6	**15,2**	**4,9**	1,8	0,6	3,2	15,7	**46,9**	3,2	153,5
GEOP	241	1189	72	1	171	257	122	398	25	195	2671
Moy.	11,4	56,6	3,4	0,05	**8,1**	**12,2**	5,8	18,9	1,1	**9,2**	127,1
Total	545	2430	452	124	216	273	202	792	1198	277	6509
Couv.%	8,37	**37,33**	6,94	1,91	3,32	4,19	3,10	12,17	18,41	4,26	100%

TABLE 1 – Distributions des labels logiques au sein de LING et GEOP

qui le précède et deux items contigus sont coordonnés. Deuxièmement, les structures multi-échelles du corpus ANNODIS ont été ajoutées manuellement dans ces arbres. Ainsi, deux paragraphes introduits par des marqueurs d'intégration linéaire peuvent jouer le rôle d'item au sein d'une structure multi-échelle et être subordonnés à un paragraphe apparu précédemment dans le texte endossant le rôle d'amorce.
Le travail de cette troisième étape s'est concentré sur l'étude des relations de dépendance entre les unités logiques élémentaires suivantes : titre, paragraphe, item, citation, référence bibliographique et, enfin, byline. Il nous est apparu que, bien que les en-têtes, les pieds de page et les notes de bas de page participent à la structure organisationnelle des documents, ces unités pouvaient faire l'objet de traitements différenciés, car hors du corps de texte.
Au terme de cette étape, une différence significative apparaît (χ^2 avec α à 0,001) : LING présente des structures organisationnelles beaucoup plus riches avec de nombreuses relations de subordination et de coordination (Tableau 2). Cette différence avec GEOP s'explique encore une fois par le caractère linguistique, pour les subordinations entre paragraphes et exemples linguistiques, et académique, pour les coordinations entre références bibliographiques.

	subordination	coordination	Total
LING	714	2467	3181
Moy.	**28,56**	**98,68**	127,24
GEOP	391	1029	1420
Moy.	18,62	49	67,62
Total	1105	3496	4601
Couv.%	0,24	0,76	100%

TABLE 2 – Distributions des relations de dépendance au sein de LING et GEOP

Notons que, conformément à la licence Creative Commons By-NC-SA 3.0 de ANNODIS, les versions enrichies de LING et GEOP décrites dans cet article sont partagées selon les mêmes conditions [6].

5 Deux tâches pour la détection de la structure organisationnelle

Afin de construire l'arbre correspondant à la *structure organisationnelle* des documents, nous avons décomposé le problème en deux tâches séquentielles :

- **Tâche 1** : labellisation des *blocs visuels* issus de LA-PDFText avec les labels des unités logiques élémentaires (décrites en Section 4) au moyen de marqueurs visuels. Chaque séquence de labels pour un document obtenue en sortie est alors considéré comme la structure logique de surface de ce document.
- **Tâche 2** : construction avec un parseur *shift-reduce* de l'arbre en dépendance reliant les unités logiques élémentaires par les relations *subordination* et *coordination* au moyen de marqueurs visuels et lexicaux, et des labels issus de la Tâche 1. L'arbre en dépendance en sortie est alors considéré comme la structure logique profonde.

Ces deux tâches utilisent respectivement un *Conditional Random Fields* (CRFs), proposé par (Lafferty *et al.*, 2001), et une *régression logistique multinomiale*, introduite en TAL par (Berger *et al.*, 1996) sous le nom de *Maximum d'Entropie* (MaxEnt). Ces deux modèles sont des modèles discriminants, exponentiels et probabilistes qui permettent d'associer à une observation x sa probabilité d'appartenance à un label y, noté $p(y|x)$. Ces deux modèles sont proches, toutefois une différence réside dans le paradigme d'apprentissage ; le CRFs est un modèle graphique et apprend sur des séquences

6. http://github.com/jfaucon/corpus-LING_GEOP

d'observations $X = (x_1, x_2, \ldots, x_t)$ où il peut exister une dépendance statistique entre les labels $Y = (y_1, y_2, \ldots, y_t)$ associés à chaque séquence. Le MaxEnt modélise la probabilité conditionnelle pour une paire (x, y) unique.

Tâche 1. Nous utilisons un CRFs linéaire pour modéliser les dépendances entre les labels logiques y_t et y_{t-1} de deux blocs visuels contigus, ainsi que des informations locales riches sur chaque bloc. La prise en compte des dépendances entre labels est particulièrement adaptée pour capturer et généraliser l'ordre des blocs dans les séquences de documents. Par exemple, un titre est souvent suivi d'un paragraphe, et une référence bibliographique se situe généralement en fin de séquence. Dans sa version du premier ordre, un *champ conditionnel aléatoire* (CRF) prend la forme :

$$p_\theta(y|x) = \frac{1}{Z_\theta(x)} \exp\left(\sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, x_t)\right) \tag{1}$$

Tâche 2. Nous utilisons conjointement un parseur *shift-reduce* et un MaxEnt pour modéliser la probabilité conditionnelle d'une paire (x, y) où x est une transition entre deux *unités logiques élémentaires* contiguës et y est l'ensemble {subordination, coordination, $\varnothing$}. Le MaxEnt est un modèle plus adapté aux situations où les distributions sont asymétriques (Malouf, 2002), comme c'est le cas avec la distribution des relations de dépendance. Le MaxEnt prend la forme :

$$p_\theta(y|x) = \frac{1}{Z_\theta(x)} \exp\left(\sum_{k=1}^{K} \theta_k f_k(y, x)\right) \tag{2}$$

Dans les deux modèles, $Z_\theta(x)$ est une constante de normalisation qui assure que la somme des probabilités égale 1, ainsi $Z_\theta(x)$ assure pour le CRF $\sum_y p_\theta f(y_t, y_{t-1}, x) = 1$ et pour le MaxEnt $\sum_y p_\theta f(y, x) = 1$. À chacun des K traits f_k est associé un paramètre θ_k qui donne un poids quant à l'appartenance de x à y.
Théoriquement, le problème dual du MaxEnt, où il s'agit de choisir sous des contraintes calculées à partir des traits la distribution maximisant l'entropie, est semblable à celui du CRFs qui maximise la somme des entropies sous des contraintes calculées identiquement (Ganapathi *et al.*, 2008) [7]. Dans la pratique, l'estimation du vecteur de paramètres θ s'effectue au travers de la maximisation, sans contraintes, de la log-vraisemblance pénalisée sur le corpus d'apprentissage T $(x^{(i)}, y^{(i)})_{i=1}^{N}$. Ainsi, dans les deux modèles, l'estimateur $\hat{\theta}$ est obtenu en (3) par la maximisation de $\mathcal{L}(\theta)$ (4) :

$$\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmax}}\, \mathcal{L}(\theta) \tag{3}$$

$$\mathcal{L}(\theta) = \frac{1}{N}\left[\sum_{n=1}^{N} \tilde{p}(x^{(i)}, y^{(i)})\ \log\ p_\theta(y^{(i)}|x^{(i)})\right] - \alpha \cdot R(\theta) \tag{4}$$

où $\tilde{p}(x^{(i)}, y^{(i)})$ est la fréquence empirique observée dans T, $R(\theta)$ est un facteur de régularisation et α son coefficient. La propriété de convexité de $\mathcal{L}(\theta)$ assure qu'un *local optimum* est aussi le *global optimum*. Ainsi, une solution unique existe et différents algorithmes itératifs assurent la convergence vers cette dernière. Toutefois, le CRFs nécessite une phase d'inférence à chaque itération pour calculer le gradient de $\mathcal{L}(\theta)$, lié au calcul de la dépendance entre labels. Cette inférence est généralement réalisée avec l'algorithme *forward-backward*, qui présente néanmoins un coût élevé.
Dans notre travail, nous utilisons l'implémentation *crf.sourceforge.net* [8], que nous avons légèrement modifiée pour le support du MaxEnt et la lisibilité des sorties. Pour les deux modèles, nous régularisons $\mathcal{L}(\theta)$ selon la norme L_2 en posant $R(\theta) = \sum_{k=1}^{K} \theta_k^2$. Enfin, nous optimisons $\mathcal{L}(\theta)$ avec l'algorithme LM-BFGS, recommandé pour le MaxEnt par (Malouf, 2002) et pour le CRFs par (Sha & Pereira, 2003).

5.1 Tâche 1 : Labellisation des blocs visuels en unités logiques élémentaires

Pour cette tâche, l'objectif est d'attribuer un label logique aux blocs visuels issus de LA-PDFText sur la base de leurs propriétés de réalisation dans le document. L'hypothèse est double : (i) il est possible d'attribuer un label (e.g : titre, paragraphe, item, etc.) à un bloc visuel à partir de sa mise en forme et (ii) les documents présentent généralement leurs unités logiques selon une séquence générique (e.g : généralement un titre est suivi d'un paragraphe, etc.).

Par conséquent, nous avons défini deux familles de traits : les *traits locaux* qui portent sur les informations locales d'un bloc visuel et les *traits de séquence* qui donnent des informations relatives à la position d'un bloc visuel dans la séquence du document en cours d'apprentissage. Le tableau 3 propose un aperçu synthétique des traits de ces deux familles.

7. Ce problème dual est celui du modèle *Maximum Entropy Markov Models* (HMMs) (McCallum *et al.*, 2000) que le CRFs partage.
8. `http://crf.sourceforge.net`

Les traits locaux se veulent génériques. Ils utilisent des valeurs relatives à chaque document (e.g : une police apparaît majoritairement dans le document, il y a une indentation à gauche, etc.) à la place des valeurs absolues (e.g : une police de taille 10, un retrait de 40 pixels, etc.). Cette manière de procéder permet de se détacher en partie des conventions de mise en forme, qui varient selon le support, le moyen de production ou le mode de divulgation des documents, et d'éviter d'induire des biais dans l'apprentissage. Dans la pratique, ce travail de généralisation des traits nécessite une phase de pré-traitement pour chaque document. Cette phase calcule le mode des variables discrètes (e.g : marges, tailles des polices, etc.) et nominales telles que le style des polices (e.g : Times New Roman, Arial, etc.) ou la présence d'emphase.

Familles	Traits	Informations capturées
Traits locaux	*marges*	Indentation à droite ou à gauche, centrage des blocs, absence d'indentation, etc.
	polices	Présence d'emphases (gras ou italique), taille de la police, etc.
	typographie	Présence de puces, de tirets, de numérotation, d'un « ; » ou « , » en fin de bloc, etc.
	position	Position verticale dans la page (haut, bas) et horizontale (droite, gauche).
	ratios	Ratios de la surface sur la taille de la police, de longueur sur la largeur, etc.
Traits de séquence	*bigrammes*	Considère le label y attribué à l'unité qui précède dans la séquence du document.
	start/end	Présence du bloc en début ou en fin de document.
	contraste	Rupture avec le bloc qui précède (taille, type de police, indentation, etc.).

TABLE 3 – Traits pour la Tâche 1

Les traits *ratios* sont une généralisation statistique de plusieurs caractéristiques locales au sein d'une même formule. Les distributions de valeurs de ces *ratios* sont discrétisées par un découpage en déciles. À chaque décile est associé un trait binaire qui renvoie vrai si le ratio du bloc visuel courant appartient à ce décile.

5.2 Tâche 2 : Construction de l'arbre en dépendance

Pour cette tâche, deux objectifs sont poursuivis : (i) attribuer une relation de dépendance entre deux unités logiques élémentaires et (ii) construire l'arbre correspondant à l'ordonnancement de ces unités au sein de chaque document. Les propriétés hiérarchiques de la représentation de la structure organisationnelle (Section 3) offrent la possibilité de réaliser ces deux objectifs simultanément avec des techniques comparables à celles utilisés en analyse syntaxique.
Nous avons utilisé l'adaptation de l'algorithme *shift-reduce* proposée par (Hernandez & Grau, 2005) pour les énoncés, qui prend en compte les relations de *subordination* et de *coordination*. Le principe de cet algorithme à pile est de parcourir la séquence des unités logiques élémentaires, de gauche à droite, et de chercher le point d'attachement optimal à gauche pour chaque bloc. À chaque étape de la construction de l'arbre, au minimum deux unités sont simultanément inspectées.

Le MaxEnt est entraîné sur trois classes ; la subordination, la coordination et l'absence de relation (notée $\varnothing$). Si une relation de subordination est détectée, l'algorithme descend dans la structure du document (e.g : un paragraphe et un item). Si une relation de coordination est détectée, l'algorithme reste au même niveau dans le document (e.g : deux paragraphes). Enfin, si aucune relation n'est trouvée, l'algorithme remonte dans la structure du document (e.g : un paragraphe et un titre).

Les traits utilisent (i) des informations visuelles (typographiques et dispostionnelles), (ii) des informations lexicales correspondant aux marqueurs d'intégration linéaire, (iii) les labels des unités logiques élémentaires et, enfin, (iv) des informations liés au parallélisme (visuel et lexical) entre unités logiques. Le tableau 4 présente synthétiquement ces traits. Les marqueurs liés aux traits *visuels* sont obtenus de manière similaire à la Tâche 1 (Section 5.1). Les traits *lexique* utilisent une liste prédéfinie de marqueurs d'intégration linéaire. Les traits *labels* et *parallélisme* reposent sur l'hypothèse que nous disposons des résultats produits en sortie de la Tâche 1.

Traits	Informations capturées
visuels	Présence d'indentation, de tirets, de puces, de « : », etc.
lexique	Présence de marqueurs d'intégration linéaire (e.g : *Premièrement*, *Deuxièmement*, etc.).
labels	Paires de labels (e.g : titre-paragraphe, item-item, paragraphe-item, etc.) et égalité de labels
parallélisme	Paragraphe entre deux items visuellement identiques, deux items mais différents , etc.

TABLE 4 – Traits pour la Tâche 2

6 Évaluation

Pour les deux tâches, nous avons procédé à validation croisée (k =10) et présentons les résultats en termes d'exactitude.

Tâche 1. Pour évaluer cette tâche, nous posons la *baseline naïve* consistant à classer tous les blocs visuels en paragraphes, qui forment la classe majoritaire dans LING et GEOP (Section 4). Pour chaque corpus, nous avons effectué l'évaluation selon deux configurations : (i) avec les *traits locaux* seuls (indices typographiques et dispostionnels) et, ensuite, (ii) avec les traits locaux adjoints aux *traits de séquence* (propres au CRFs). Les résultats sont reportés dans le tableau 5.

Approches	LING	GEOP	LING_GEOP
Traits locaux	78,37%	79,97%	73,63%
+ Traits de séquence	**87,18%**	**82,39%**	**80,46%**
Baseline naïve	32,33%	44,51%	37,33%

TABLE 5 – Évaluation pour la labellisation en unités logiques élémentaires (Tâche 1)

Dans la première configuration, les résultats pour LING montrent une difficulté à classer les blocs visuels, avec un léger recul par rapport à GEOP (Δ1,60%). Ce taux bas s'explique notamment par les nombreux exemples linguistiques au sein de LING (Section 4). Ces unités, considérés comme items dans la structure logique de surface présentent des caractéristiques visuelles différentes des items « classiques ». Leur rôle métadiscursif de *citation* implique une indépendance de leur contexte d'apparition. Par exemple, elles ne suivent pas les conventions typographiques des items (e.g : un « ; » en milieu d'énumération et un « . » à la fin) et leur numérotation suit leur ordre d'énonciation dans le document.

Dans la deuxième configuration, la prise en compte de la structure du document permet de palier en partie les variations locales des unités. Cela se traduit par des augmentations significatives (test de Wilcoxon avec α à 0,05) par rapport à la première configuration dans LING (p < 0,01) et GEOP (p = 0,023). Toutefois, pour GEOP, cette amélioration (Δ2,24%) n'est pas aussi élevée que pour LING (Δ8,81%).
Cette différence s'explique notamment par le caractère moins structuré et visuellement hétérogène de GEOP induisant une distribution différente des labels lors de l'apprentissage. Dans les deux corpus, les paragraphes, largement majoritaires, sont correctement classés : F-score de 93,47 pour GEOP et de 90,88 pour LING. Cependant, les nombreuses unités de la classe *autres* (figures, tableaux, etc.) rompent régulièrement la séquence de labels dans GEOP. Ainsi, pour les items, leur nombre restreint et ces variations induisent une diminution (F-score de 26,47 face à 67,58 dans LING). Le même phénomène apparaît également avec les titres de niveau 2 (F-score de 53,17 face à 95,45 dans LING).

Afin de diminuer les variations de distributions dans les corpus d'apprentissage, une évaluation a été faite sur les deux corpus pris conjointement (LING_GEOP). Cette approche avec traits de séquence montre aussi une hausse (Δ6,83%) par rapport aux traits locaux. La figure 3 présente les courbes d'apprentissage avec les traits de séquence sur les trois corpus. Pour obtenir ces courbes, une validation croisée (k=10) à été exécutée pour chaque n de documents choisis aléatoirement dans les 9 ensembles restants. Les résultats semblent indiquer qu'un agrandissement du corpus améliore les scores.

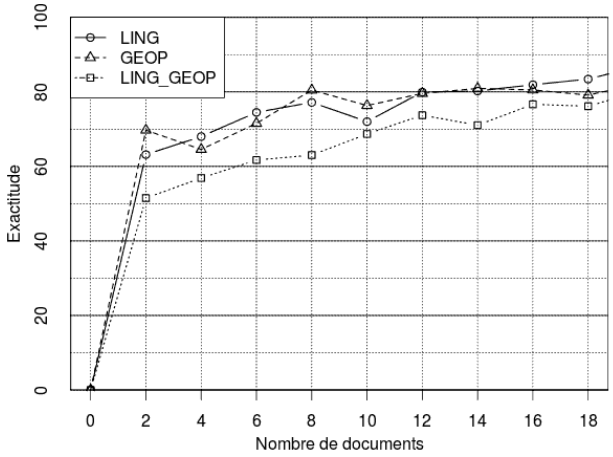


FIGURE 3 – Courbes d'apprentissage pour LING, GEOP et LING_GEOP (Tâche 1)

Tâche 2. Pour l'évaluation de cette tâche, nous proposons une *baseline naïve* consistant à classer aléatoirement les relations de subordination et de coordination liant deux unités logiques. L'approche par *traits* est celle décrite en section 5.2. Nous proposons une comparaison avec une approche par *grammaire formelle* décrivant les règles *a priori* d'organisation d'un document (utilisée pour la construction du corpus en section 4). Les résultats sont reportés dans le tableau 6.

Approches	LING	GEOP	LING_GEOP
Traits	96,41%	**98,45%**	**97,23%**
Grammaire	**96,54%**	98,30%	97,08%
Baseline naïve	40,21%	41,03%	39,79%

TABLE 6 – Évaluation pour la construction de l'arbre en dépendance (Tâche 2)

Dans l'approche par traits, la différence entre LING et GEOP (Δ2,04%) s'explique par la structuration complexe, en termes de dépendances, au sein de LING. Certains documents dans LING montrent des niveaux d'imbrications très profond, notamment par l'utilisation d'exemples linguistiques imbriqués dans des énumérations de définitions. Cela se traduit par des scores différents pour les subordinations (F-score de 91,99 pour LING et de 97,15 pour GEOP), tandis que ceux obtenus pour les coordinations restent relativement équivalents (F-score de 97,15 pour LING et de 98,93 pour GEOP).

Les scores du tableau 6 ne montrent pas de différences significatives entre l'approche par traits et la grammaire (test de Wilcoxon avec α à 0,05). Deux raisons expliquent cela. Premièrement, les relations entre les unités suivent majoritairement les règles définies dans la grammaire. Seuls certains cas (e.g : imbrications profondes, dépendances de longue distance, etc.) permettent de distinguer grammaire et traits. Deuxièmement, cette asymétrie dans la distribution des cas (respectent vs ne respectent pas la grammaire) induit un phénomène d'apprentissage de la grammaire et non des traits considérés comme discriminants (e.g : deux items contigus visuellement différents, un paragraphe indenté, etc.).

Par conséquent, pour mesurer l'apport de l'approche par traits, nous proposons d'évaluer uniquement les cas où la relation entre deux unités diffère de l'ordonnancement *a priori*, c'est-à-dire lorsque la grammaire ne peut fournir la réponse correcte. Les résultats de cette stratégie *traits sur erreurs grammaire* montrent un léger gain qui reste stable au travers des corpus (Tableau 7). Le pendant de cette stratégie consiste à évaluer les traits hors de ces cas. Ces résultats sont ceux de *traits hors erreurs grammaire*, où sont reportés 20 erreurs pour LING, 2 pour GEOP et 12 pour LING_GEOP.

Stratégies	LING	GEOP	LING_GEOP
Traits sur erreurs grammaire	14,54% (16/110)	16,66% (4/24)	14,17% (19/134)
Traits hors erreurs grammaire	99,34% (3051/3071)	99,85% (1394/1396)	99,73% (4455/4467)

TABLE 7 – Deux stratégies pour évaluer les traits face à la grammaire (Tâche 2)

7 Discussion

Les labels logiques utilisés dans les Tâches 1 et 2, ainsi que l'adoption d'une représentation arborée sont des éléments partagés avec l'*Analyse de Documents*. Toutefois, l'objectif de notre travail a été de proposer une représentation en lien avec le discours, mais restant adaptée à l'analyse de documents. Pour cela, il a été choisi de travailler sur le typage des contenus uniquement textuels et une réflexion a été menée sur la différence entre mise en forme et rôle métadiscursif des unités (e.g : un bloc visuel formaté comme paragraphe n'endosse pas toujours le rôle de paragraphe). Les relations de dépendance proposées permettent de représenter cette différence et ouvrent la voie à l'identification de phénomènes discursifs complexes (e.g : énumérations de définitions, etc.). Ces choix, qui ont nécessité l'enrichissement de corpus annotés discursivement, rendent difficile l'utilisation d'outils classiques d'analyse logique. Toutefois, une comparaison externe sur des corpus partageant les mêmes labels textuels est une perspective immédiate à la Tâche 1.

Notre méthode a été testée sur des corpus de natures différentes (une mise en forme unifiée et une structure complexe pour LING, un formatage hétérogène et une structure linéaire pour GEOP). Les résultats obtenus pour les deux tâches sont relativement corrects. Toutefois, des expériences consistant à utiliser en séquence les deux modules ont montré que la Tâche 2 était très sensible au bruit. Ceci constitue pour l'instant un aspect limitatif de notre solution.

Également pour la Tâche 2, il apparaît que les traits lexicaux n'apportent qu'un léger gain par rapport à une approche déterministe. Deux raisons expliquent cette limite. Premièrement, le grain choisi pour l'analyse se limite aux blocs visuellement indépendants et empêche de traiter les cas où les marqueurs d'intégration linéaire sont intra-paragraphiques. Or, ce type de construction est fréquent dans le corpus ANNODIS. Une perspective consistera à travailler avec une granularité plus fine, rapprochant nos travaux de ceux de (Hernandez & Grau, 2005), mais en gardant les marqueurs visuels. Deuxièmement, cette limite s'explique aussi par la variabilité du lexique qui est fonction de la langue et du corpus. Pour améliorer le système, il est nécessaire soit d'étendre les listes données en entrée, ce qui présente un coût, soit d'approcher la tâche de manière plus générique. C'est dans cette dernière direction que nous pensons poursuivre nos recherches. Des traits incorporant des informations syntaxiques pourraient être discriminants. Par exemple, pour l'énumération, il apparaît couramment que la proposition de l'amorce soit liée syntaxiquement aux propositions des items.
Notons que d'autres travaux ont utilisé conjointement mise en forme visuelle et contenu lexical (Klink *et al.*, 2000; Ratté *et al.*, 2007), mais sans proposer une solution traitant les structures imbriquées sur plusieurs niveaux.

8 Conclusion

La contribution de notre approche réside dans la construction automatique de la structure organisationnelle de documents à partir de marqueurs métadiscursifs de nature typographique, dispositionnelle et lexicale. Cette structure est représentée par un arbre en dépendance agençant les unités logiques selon leur label et le rôle métadiscursif qu'elles endossent.
Les perspectives générales de ce travail vont dans deux directions. Premièrement, il est envisagé d'étendre notre approche aux documents numériques tels que les pages HTML de Wikipédia. Ces documents présentent une structuration différente où l'aspect discursif est souvent suppléé par des marqueurs visuels (Bush, 2003). Leur balisage originel permettra de les faire entrer dans le système directement au sein de la Tâche 2. Deuxièmement, l'utilisation de modèles d'apprentissage non-supervisé (clustering) est considérée dans la Tâche 1 afin de ne pas faire d'hypothèse sur les labels logiques. Il s'agira de regrouper les unités de mêmes mises en forme afin d'en prédire le rôle métadiscursif commun dans un second temps.

Références

AIELLO M., MONZ C., TODORAN L. & WORRING M. (2002). Document understanding for a broad class of documents. *International Journal on Document Analysis and Recognition*, **5**(1), 1–16.

ASHER N. (1993). *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers, Dordrecht.

BATEMAN J., KAMPS T., KLEINZ J. & REICHENBERGER K. (2001). Towards constructive text, diagram, and layout generation for information presentation. *Computational Linguistics*, **27**(3), 409–449.

BERGER A., PIETRA V. & PIETRA S. (1996). A maximum entropy approach to natural language processing. *Computational linguistics*, **22**(1), 39–71.

BOSSARD A. (2009). Une approche mixte-statistique et structurelle-pour le résumé automatique. In *Actes de la 16e Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2009)*.

BOUAYAD-AGHA N., POWER R. & SCOTT D. (2000). Can text structure be incompatible with rhetorical structure ? In *Proceedings of the first international conference on Natural language generation-Volume 14*, p. 194–200 : Association for Computational Linguistics.

BUSH C. (2003). Des déclencheurs des énumérations d'entités nommées sur le web. *Revue québécoise de linguistique*, **32**(2), 47–81.

CHAROLLES M. (1997). L'encadrement du discours. In *Cahier de Recherche Linguistique*, volume 6. Université de Nancy.

CHOI F. Y. Y. (2002). *Content-based Text Navigation*. PhD thesis, the University of Manchester.

COUTO J., FERRET O., GRAU B., HERNANDEZ N., JACKIEWICZ A., MINEL J.-L. & PORHIEL S. (2004). Régal, un système pour la visualisation sélective de documents. *Revue d'intelligence artificielle*, **18**(4), 481–514.

FAUCONNIER J., KAMEL M., ROTHENBURGER B. & AUSSENAC-GILLES N. (2013). Apprentissage supervisé pour l'identification de relations sémantiques au sein de structures énumératives parallèles. In *Actes de la 20e Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2013)*, p. 132–145.

GANAPATHI V., VICKREY D., DUCHI J. & KOLLER D. (2008). Constrained approximate maximum entropy learning of markov random fields. In *Conference on uncertainty in artificial intelligence (UAI)*.

HARRIS Z. (1971). Structures mathématiques du language. *Dunod. Paris, France*.

HERNANDEZ N. & GRAU B. (2005). Détection automatique de structures fines de texte. In *Actes de la 12e Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2005)*.

HO-DAC L.-M., PÉRY-WOODLEY M.-P. & TANGUY L. (2010). Anatomie des structures énumératives. In *Actes de la 17e conférence sur le Traitement Automatique des Langues Naturelles (TALN 2010)*.

KLINK S., DENGEL A. & KIENINGER T. (2000). Document structure analysis based on layout and textual features. In *Proc. of International Workshop on Document Analysis Systems, DAS2000*, p. 99–111 : Citeseer.

LAFFERTY J., MCCALLUM A. & PEREIRA F. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. *Department of Computer & Information Science, University of Pennsylvania*.

LUC C. & VIRBEL J. (2001). Le modèle d'architecture textuelle : fondements et expérimentation. *Verbum*, (1), 103–123.

LÜNGEN H., BÄRENFÄNGER M., HILBERT M., LOBIN H. & PUSKÁS C. (2010). Discourse relations and document structure. *Linguistic modeling of information and markup languages*, **1**, 97–123.

MALOUF R. (2002). A comparison of algorithms for maximum entropy parameter estimation. In *proceedings of the 6th conference on Natural language learning*, p. 1–7 : Association for Computational Linguistics.

MANN W. & THOMPSON S. (1988). Rhetorical structure theory : Toward a functional theory of text organization. *Text*, **8**(3), 243–281.

MAO S., ROSENFELD A. & KANUNGO T. (2003). Document structure analysis algorithms : a literature survey. In *Electronic Imaging 2003*, p. 197–207 : International Society for Optics and Photonics.

MARCU D. (2006). Automatic discourse parsing. In K. BROWN, Ed., *Encyclopedia of Language and Linguistics*. Elsevier, 2nd edition.

MCCALLUM A., FREITAG D. & PEREIRA F. (2000). Maximum entropy markov models for information extraction and segmentation. In *ICML*, p. 591–598.

MYERS E. W. (1986). Ano (nd) difference algorithm and its variations. *Algorithmica*, **1**(1-4), 251–266.

NUNBERG G. (1990). *The linguistics of punctuation*. Number 18. CSLI Publications.

PAASS G. & KONYA I. (2012). Machine learning for document structure recognition. In A. MEHLER, K.-U. KÜHNBERGER, H. LOBIN, H. LÜNGEN, A. STORRER & A. WITT, Eds., *Modeling, Learning, and Processing of Text Technological Data Structures*, volume 370 of *Studies in Computational Intelligence*, chapter Part V : Document Structure Learning, p. 221–247. Springer.

POWER R., SCOTT D. & BOUAYAD-AGHA N. (2003). Document structure. *Computational Linguistics*, **29**(2), 211–260.

PÉRY-WOODLEY M.-P., AFANTENOS S. D., HO-DAC L.-M. & ASHER N. (2011). La ressource annodis, un corpus enrichi d'annotations discursives. *Traitement Automatique des Langues (TAL)*, **52**(3), 71–101.

PÉRY-WOODLEY M.-P. & SCOTT D. (2006). Computational approaches to discourse and document processing. *TAL*, **47**(2), 7–19.

RAMAKRISHNAN C., PATNIA A., HOVY E. H. & BURNS G. (2012). Layout-aware text extraction from full-text pdf of scientific articles. *Source code for biology and medicine*, **7**(1).

RATTÉ S., NJOMGUE W. & MÉNARD P.-A. (2007). Highlighting document's structure. *International Journal of Computer Science & Engineering*, **1**(2).

SHA F. & PEREIRA F. (2003). Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, p. 134–141 : Association for Computational Linguistics.

SORIN L., MOJAHID M., AUSSENAC-GILLES N. & LEMARIÉ J. (2013). Improving the accessibility of digital documents for blind users : contributions of the textual architecture model. In M. A. CONSTANTINE STEPHANIDIS, Ed., *Universal Access in Human-Computer Interaction. Applications and Services for Quality of Life*, p. 399–407. Springer.

TOKUYASU T. A. & CHOUB P. A. (2001). Turbo recognition : a statistical approach to layout analysis. In *Proceedings of SPIE*, volume 4307, p. 124.

VERGEZ-COURET M., BRAS M., PREVOT L., VIEU L., ATTALAH C. *et al.* (2011). The discourse contribution of enumerative structures involving 'pour deux raisons'. In *Procceedings of Constraints in Discourse*.

VIRBEL J., LUC C., SCHMID S., CARRIO L., DOMINGUEZ C., PÉRY-WOODLEY M.-P., JACQUEMIN C., MOJAHID M., BACCINO T. & GARCIADEBANC C. (2005). Approche cognitive de la spatialisation du langage. de la modélisation de structures spatio-linguistiques des textes à l'expérimentation psycholinguistique : le cas d'un objet textuel, l'énumération. In C. THINUS-BLANC & J. BULLIER, Eds., *Agir dans l'Espace*, chapter 12, p. 233–254. Paris : Editions de la MSH.

Jugement exact de grammaticalité d'arbre syntaxique probable

Jean-Philippe Prost
LIRMM, CNRS – Université Montpellier 2, 161 rue Ada, 34090 Montpellier, France
Prost@lirmm.fr

Résumé. La robustesse de l'analyse probabiliste s'obtient généralement au détriment du jugement de grammaticalité sur la phrase analysée. Les analyseurs comme le Stanford Parser, ou les Reranking Parsers ne sont, en effet, pas capables de dissocier une analyse probable grammaticale d'une analyse probable erronée, et ce qu'elle porte sur une phrase elle-même grammaticale ou non. Dans cet article nous montrons que l'adoption d'une représentation syntaxique basée sur la théorie logique des modèles, accompagnée d'une structure syntaxique classique (par exemple de type syntagmatique), est de nature à permettre la résolution exacte de différents problèmes tels que celui du jugement de grammaticalité. Afin de démontrer la praticité et l'utilité d'une alliance entre symbolique et stochastique, nous nous appuyons sur une représentation de la syntaxe par modèles, ainsi que sur une grammaire de corpus, pour présenter une méthode de résolution exacte pour le jugement de grammaticalité d'un arbre syntagmatique probable. Nous présentons des résultats expérimentaux sur des arbres issus d'un analyseur probabiliste, qui corroborent l'intérêt d'une telle alliance.

Abstract. The robustness of probabilistic parsing generally comes at the expense of grammaticality judgment – the grammaticality of the most probable output parse remaining unknown. Parsers, such as the Stanford or the Reranking ones, can not discriminate between grammatical and ungrammatical probable parses, whether their surface realisations are themselves grammatical or not. In this paper we show that a Model-Theoretic representation of Syntax alleviates the grammaticality judgment on a parse tree. In order to demonstrate the practicality and usefulness of an alliance between stochastic parsing and knowledge-based representation, we introduce an exact method for putting a binary grammatical judgment on a probable phrase structure. We experiment with parse trees generated by a probabilistic parser. We show experimental evidence on parse trees generated by a probabilistic parser to confirm our hypothesis.

Mots-clés : Jugement de grammaticalité, syntaxe par modèles, Grammaires de Propriétés, analyse syntaxique probabiliste.

Keywords: Grammaticality judgement, Model-Theoretic Syntax, Property Grammar, probabilistic syntactic parsing.

1 Introduction : jugement automatique de grammaticalité

Les analyseurs syntaxiques les plus performants du moment [1] reposent généralement sur des méthodes d'approximation probabiliste, qui leurs confèrent une très grande robustesse – au sens habituellement entendu en matière d'analyse syntaxique, à savoir l'aptitude à générer une analyse, fût-elle partielle, pour n'importe quelle entrée. Cependant cette aptitude s'accompagne d'une absence de jugement de grammaticalité quant à la phrase. Cette absence tient au fait qu'une analyse optimale, même grammaticale, ne se voit jamais attribuée de probabilité maximale, ce qui interdit le jugement binaire exact attendu en matière de grammaticalité. Or selon le contexte applicatif la méconnaissance de la grammaticalité est préjudiciable. Cette carence est soulignée dans divers travaux concernant, par exemple, la génération automatique (Wan *et al.*, 2005), la traduction automatique statistique (Zwarts & Dras, 2008), l'évaluation de compétence en langue étrangère, ou encore bien sûr la correction grammaticale (Bender *et al.*, 2004; Tetreault & Chodorow, 2008). Ces applications pratiques conduisent à mettre en œuvre différentes méthodes pour établir un jugement, parmi lesquelles certaines sont symboliques et d'autres probabilistes.

Dans cet article nous argumentons qu'une représentation de la syntaxe du langage naturel basée sur la théorie logique des modèles permet de concevoir le jugement de grammaticalité comme un procédé exact de vérification de modèle. Dans une première partie nous commençons par faire un tour d'horizon de la littérature sur la question, et présentons rapidement les

1. D'après le wiki ACL, `http://aclweb.org/aclwiki/index.php?title=Parsing_(State_of_the_art)` (au 7 mars 2014). Dernière mise-à-jour le 28 octobre 2013.

méthodes existantes, tant exactes que stochastiques. Dans une deuxième partie nous exposons quelques fondamentaux sur la représentation de la syntaxe par modèles, et montrons en quoi une telle représentation, combinée avec une génération probabiliste de structures syntaxiques, permet de résoudre aisément le problème de la grammaticalité d'une structure. Dans une troisième partie nous détaillons le procédé de vérification de modèle qui permet le jugement d'une structure. Nous présentons ensuite dans une quatrième partie l'expérimentation que nous avons menée sur le corpus Sequoia (Candito & Seddah, 2012) pour mettre à l'épreuve ce procédé. Nous concluons enfin en dernière partie, où nous discutons quelques perspectives intéressantes, notamment en matière d'amélioration qualitative des analyseurs syntaxiques probabilistes.

2 État de l'art

Les méthodes par approximation probabiliste Une solution couramment envisagée pour juger la grammaticalité d'une phrase consiste à faire appel à la classification stochastique binaire d'arbres syntaxiques probables. Foster *et al.* (2008); Wagner (2012) dans un cadre de correction grammaticale (focalisé sur des types d'erreur spécifiques), ou encore Wong & Dras (2011) dans un cadre plus général, combinent plusieurs grammaires probabilistes [2] pour l'entraînement d'un analyseur stochastique, pou ensuite analyser différents corpus, grammaticaux et erronés. Le résultat est utilisé pour extraire de nouveaux traits d'apprentissage destinés au classificateur de grammaticalité. Le jugement s'effectue alors autour d'un seuil de probabilité, qu'il est nécessaire de calibrer expérimentalement. Tous ces travaux justifient leur approche par le fait qu'ils se heurtent à la "trop grand robustesse" (dixit Wong & Dras) des analyseurs stochastiques utilisés [3], puisque dans leur configuration initiale toutes les phrases candidates à l'analyse [4] reçoivent un arbre probable.

Notons que l'hypothèse commune sous-jacente concernant la corrélation entre probabilité et grammaticalité est, en quelque sorte, déjà présente chez Charniak & Johnson (2005), qui s'appuient sur un classificateur à Maximum d'Entropie pour réordonner les analyses initiales du parseur de Charniak selon de nouvelles probabilités.

Les méthodes symboliques Les travaux qui s'appuient sur des méthodes symboliques de résolution, notamment en matière de correction grammaticale, font le plus souvent appel à des stratégies ad hoc, centrées autour de classes d'erreurs plus ou moins communément observées. L'utilisation de grammaires d'erreurs, et de techniques de recouvrement d'échec d'analyse sont parmi les stratégies les plus fréquentes. Ces approches exactes présentent souvent l'avantage d'être redoutablement plus efficaces que leurs concurrentes, mais sur des classes de problèmes très spécifiques. Néanmoins la généralisation se fait difficilement. Un changement de contexte nécessite souvent de réviser la méthode elle-même. On peut ainsi se demander, par exemple, si les systèmes qui sont performants sur des productions humaines resteraient tout aussi performants dans un contexte de génération automatique statistique, tel que le résumé automatique. Leur intégration semble difficile dans ces contextes où le jugement de grammaticalité n'intervient pas seulement en fin de chaîne de traitement, mais fait partie intégrante du processus en cours (généralement pour informer une étape de classement de phrases candidates (Mutton *et al.*, 2007)).

Les méthodes à base de contraintes Les processus de résolution de contraintes constituent également une approche possible, bien que relativement moins explorée. La raison principale en est que ces approches sont généralement, et rapidement, confrontées à l'écueil (assez prévisible) lié à de la gestion de l'explosion combinatoire de la taille de l'espace de recherche à parcourir.

Néanmoins, les contraintes en elles-mêmes restent un outil de représentation très puissant auquel nous ferons appel. Nous les utiliserons connectées en réseau, de manière passive, sans qu'elles n'alimentent de problème de résolution (de contraintes) propice à l'explosion. Ce réseau de contraintes jouera pour nos un rôle clé de représentation pour les connaissances linguistiques [5], et sera associé à ce qu'il est convenu d'appeler une *structure syntaxique* (de type syntagmatique pour ce qui nous concerne).

Nous formulons, en effet, ici comme hypothèse que l'utilisation d'une telle représentation, lorsqu'elle est alliée à une stratégie stochastique de parcours de l'espace de recherche, doit permettre de porter un jugement exact de grammaticalité

2. Les combinaisons discutées sont généralement parmi 3 types de grammaires : une extraite d'un corpus standard de référence (BNC, ou PTB), une issue d'un treebank artificiellement distordu, et une issue de la fusion entre le corpus standard et son pendant erroné.
3. Notamment le Stanford Parser, et le Reranking Parser.
4. à epsilon près.
5. Dans cet article nous restreignons notre champ d'investigation à la seule dimension syntaxique, mais nous pouvons raisonnablement supposer qu'une généralisation de l'approche de modélisation à d'autres dimensions linguistiques semble possible.

sur un arbre syntaxique par rapport à une grammaire donnée. Nous verrons par la suite que peu importe, en fait, la nature du processus de génération de la structure. En ce qui nous concerne, nous nous concentrons sur la modélisation de notre problème de jugement par hybridation entre méthode d'approximation probabiliste et méthode de raisonnement exact. Nous utiliserons donc, dans notre expérimentation, un analyseur probabiliste pour générer une structure.

3 Syntaxe par modèles

En matière de représentation des connaissances, les cadres formels basés sur la théorie logique des modèles ont prouvé leur meilleure adéquation que les langages formels à la représentation de ses irrégularités, pour des raisons que nous rappelons brièvement ici. Nous montrons ensuite de quelle façon ces cadres nous permettent de formaliser le problème du jugement de grammaticalité.

Représentation des connaissances par modèles Pour la question qui nous intéresse ici, ces cadres présentent l'avantage de permettre une description logique des propriétés linguistiques d'un énoncé, en dissociant notamment cette description de l'éventuelle "bonne formation" de l'énoncé observé. Les objets d'étude sont, au sens de la théorie de modèles, des *modèles* de *théories* exprimées dans un (méta-)langage formel, où une *théorie* est un ensemble d'assertions spécifiées dans un langage formel, et un *modèle* d'une théorie est une structure qui satisfait toutes les assertions de cette théorie. Cette approche s'appuie donc sur la *sémantique* associée au langage formel utilisé pour décrire les relations grammaticales existantes (ou absentes) au sein d'une phrase en langage naturel.

Syntaxe naturelle et théorie des modèles Lorsque le domaine des structures concerné porte sur la syntaxe naturelle [6], nous obtenons l'interprétation suivante :
– une théorie est un ensemble d'assertions de grammaire, spécifié par une conjonction $\Phi = \bigwedge_i \phi_i$, où chaque atome ϕ_i est une formule logique qui met en relation des éléments de la structure ;
– une structure est une structure linguistique, syntagmatique ou autre.
Une grammaire est donc une formule conjonctive, paramétrée par la structure, et une théorie est une instance de la grammaire pour une structure donnée. Pour un domaine de structures syntagmatiques, les ϕ_i sont des relations qui portent sur des constituants (e.g. *Dans un Syntagme Nominal en français, le Déterminant précède le Nom*).

Le cadre que nous utilisons est celui des Grammaires de Propriétés, introduit par Bès & Blache (1999) et Blache (2001) (GP), et pour lequel une sémantique par modèles a été formulée par Duchier *et al.* (2009). Rappelons que les GP définissent principalement 7 relations, appelées *propriétés*, sur un domaine d'arbres syntagmatiques :
– l'Obligation (pour les têtes de syntagme),
– la Constituance (pour les catégories de constituants pouvant appartenir à un même syntagme),
– l'Unicité,
– la Linéarité (pour la Précédence Linéaire, au sens de ID/LP [7]),
– l'Exigence (pour les co-occurrences requises),
– l'Exclusion (pour l'exclusion mutuelle entre constituants d'un même syntagme), et
– l'Accord.
Le tableau 1 fournit une interprétation pour la sémantique de 6 de ces relations (nous omettons volontairement l'Accord, qui nécessite l'introduction de structures de traits typés, hors propos dans cet article).

Obligation	$A : \triangle B$	au moins un fils de A est de catégorie B
Constituance	$A : S?$	la catégorie de tout fils de A doit être dans S
Unicité	$A : B!$	au plus un fils de A est de catégorie B
Linéarité	$A : B \prec C$	un fils de catégorie B précède un fils de catégorie C
Exigence	$A : B \Rightarrow C$	la présence d'un fils de catégorie B requiert celle d'un fils de catégorie C
Exclusion	$A : B \not\Leftrightarrow C$	des fils de catégories B et C sont exclus mutuellement sous un même A

TABLE 1 – Interprétation des types de propriétés usuels en GP

6. Raccourci pour "syntaxe du langage naturel".
7. ID/LP : *Immediate Dominance / Linear Precedence*.

Vérification de modèle et jugement de grammaticalité Nous commençons par poser quelques définitions, afin de fixer les notations utilisées dans ce qui suit.

Soient donc $\mathcal{S}$ un ensemble de mots dans la langue cible, et $\mathcal{E}$ un ensemble d'étiquettes dénotant des catégories morpho-syntaxiques ; un lexique est alors un sous-ensemble $V \subset \mathcal{E} \times \mathcal{S}$ (ce qui suppose implicitement que les terminaux sont des mots déjà étiquetés par des catégories morphologiques). Soit $\mathcal{P}_\mathcal{E}$ l'ensemble de toutes les propriétés possibles sur $\mathcal{E}$, une grammaire de propriétés Φ est alors définie par une paire (P_G, V_G), avec $P_G \subseteq \mathcal{P}_\mathcal{E}$.

Soit $\tau : s$ un arbre (de constituants) décoré d'étiquettes dans $\mathcal{E}$ et dont la réalisation de surface est la chaîne de mots s, et soit Φ^s une instanciation de Φ sur $\tau : s$; $\tau : s$ est un modèle pour Φ^s ssi $\tau : s$ rend Φ^s vraie. Nous notons $\tau : s \models \Phi^s$ la satisfaction de Φ^s par $\tau : s$. L'instanciation Φ^s de la grammaire Φ pour l'arbre $\tau : s$ est également appelé *réseau de contraintes*.

Definition 1. $\tau : s$ est grammatical par rapport à Φ ssi $\tau : s \models \Phi^s$.

Ramené au jugement de grammaticalité, et puisque $\Phi^s = \bigwedge_i \phi_{i,s}$, la définition 1 signifie que la structure syntaxique $\tau : s$ doit vérifier chaque instance de propriété ϕ_i de la grammaire Φ pour que la phrase s soit considérée grammaticale pour Φ. De ce fait, le jugement de grammaticalité sur un arbre syntaxique peut se ramener à un processus de vérification de modèle. Ce processus implique les étapes suivantes :
– instancier la grammaire Φ pour l'arbre syntaxique $\tau : s$,
– construire le réseau de contraintes associé Φ^s, et enfin
– vérifié la vérité de chaque formule atomique ϕ_i^s.

Génération de structures candidat-modèles Dans la mesure où une représentation par modèles est dissociée des aspects procéduraux liés à la génération de structures candidates, la stratégie de génération peut se concevoir indépendamment de la vérification. Bien qu'il soit possible de développer un processus d'inférence qui s'appuie uniquement sur la grammaire en contraintes elle-même (Maruyama, 1990; Balfourier *et al.*, 2002; Prost, 2008; Duchier *et al.*, 2010, entre autres), rien ne l'impose. Il est donc notamment possible de faire de la vérification d'arbres générés par un procédé probabiliste. La figure 1 illustre ainsi un exemple d'arbre syntaxique produit par le Stanford Parser (STP) (Green *et al.*, 2011) pour une phrase du corpus Sequoia (et donc grammaticale par définition), mais qui est jugée agrammaticale par vérification de modèle. Notons également que le type de structure syntaxique étudié par une représentation par modèles peut prendre différentes formes selon le cadre formel utilisé. Le travail pionnier de (Maruyama, 1990) utilise, par exemple, une structure en dépendances, tandis la Théorie de l'Optimalité de (Prince & Smolensky, 1993) est plus utilisée pour décrire des structures phonologiques. Les GP, pour leur part, sont principalement utilisées pour des structures syntagmatiques, même si on relève quelques travaux qui les utilisent pour l'annotation multimodale de données conversationnelles, ou l'analyse de séquences biologiques.

De la distinction entre syntaxe générative et syntaxe par modèles En comparaison avec les approches par modèles, les représentations dites génératives-énumératives [8] sont basées sur la théorie de la preuve, et s'appuient donc sur la dimension *syntaxique* de la logique. L'hypothèse forte que formulent les approches génératives est que le langage naturel peut se modéliser comme un langage formel, dont la syntaxe capture celle du langage cible. Une structure syntaxique en constituants s'obtient alors naturellement, comme la représentation graphique (au sens des graphes) de la preuve que sa réalisation de surface est une phrase qui appartient au langage cible. Or au-delà de toute considération d'école, il est aisé de voir que cette hypothèse générative exclut, de fait, toute représentation structurelle pour un énoncé qui, bien qu'exprimé en langage naturel, n'appartiendrait pas à l'ensemble $\mathcal{L}(G)$ des *mots* définis par le langage formel $\mathcal{L}$ sous-jacent [9]. En dissociant la syntaxe du langage naturel de celle du langage formel de représentation, les approches par modèles permettent une représentation plus riche, et donc une description plus fine de l'information syntaxique relative à un énoncé. Cette représentation associe la structure syntaxique à proprement parler (e.g. la structure syntagmatique), et le réseau de contraintes constitué des propriétés $\phi_{i,s}$ instanciées pour cette structure.

8. La paternité de l'appellation *Generative-Enumerative Syntax* (GES) revient à Pullum & Scholz (2001), qui discutent cette distinction entre GES et MTS (*Model-Theoretic Syntax*). En français nous parlerons de *syntaxe par modèles* pour faire référence à la MTS et aux cadres formels de représentation qui lui sont associés.

9. D'après la notation usuelle pour les grammaires syntagmatiques, où le langage $\mathcal{L}(G)$ est défini comme le n-uplet $\langle V, N, G, S \rangle$, où V est un lexique (vocabulaire terminal), N un ensemble de catégories morpho-syntaxiques (vocabulaire non-terminal), Φ un ensemble de règles de production (grammaire), et $S \in N$ un symbole de départ.

```
419:  En_effet, sept projets sur quatorze, soit la moitié, ont un financement qui
       n' est toujours pas assuré et dont le calendrier n' est pas_encore arrêté.
```

Analyse de référence dans Sequoia :

```
( (SENT (ADV En_effet) (PUNC ,)
        (NP (DET sept)
                (NC projets)
                (PP (P sur)
                    (NP (ADJ quatorze)))
                (PUNC ,)
                (COORD (CC soit)
                    (NP (DET la) (NC moitié)))) (PUNC ,)
        (VN (V ont))
        (NP (DET un)
            (NC financement)
            (Srel
                (N (PROREL qui))
                (VN (ADV n')
                    (V est)
                    (AdP (ADV toujours) (ADV pas))
                    (VPP assuré))
        (COORD (CC et)
            (Srel
                (PP (PROREL dont))
                (NP (DET le) (NC calendrier))
                (VN (ADV n') (V est) (ADV pas_encore) (VPP arrêté)))))) (PUNC .)))
```

Analyse fournie par le STP :

```
( (SENT (ADV En_effet) (PUNC ,)
        (NP (DET sept)
            (NC projets)
            (PP (P sur)
                (NP (NC quatorze)))
            (PUNC ,)
            (COORD (CC soit)
                (NP (DET la) (NC moitié)))) (PUNC ,)
        (VN (V ont))
        (NP (DET un)
            (NC financement)
            (Srel
                (NP (PROREL qui))
                (VN (ADV n')
                    (V est)
                    (AdP (ADV toujours) (ADV pas))
                    (VPP assuré))))
        (COORD (CC et)
            (Sint
                (NP (NC dont) (DET le) (NC calendrier))
                (VN (ADV n') (V est) (ADV pas_encore) (VPP arrêté)))) (PUNC .)))
```

FIGURE 1 – Exemple d'analyse syntaxique par le STP jugée agrammaticale par vérification de modèle

4 Vérification de modèle et grammaticalité

Nous l'avons vu précédemment, le jugement de grammaticalité passe par un processus de vérification de modèle. Ce processus nécessite lui-même deux ressources : une grammaire de propriétés, et une instanciation de cette grammaire pour un arbre donné, qui joue le rôle de candidat-modèle. La grammaire de propriétés que nous utilisons dérive de la grammaire hors-contexte implicite à un corpus arboré, par application de règles propres à chaque type de propriété. Nous décrivons cette dérivation dans la partie qui suit. Nous ne reprenons pas ici le processus d'instanciation de la grammaire pour un arbre donné, qui consiste simplement à unifier les constituants présents dans les propriétés avec les nœuds étiquetés de l'arbre. Dans la littérature sur les GP, l'étape d'instanciation est généralement intégrée au processus de *caractérisation*, qui couvre simultanément l'instanciation et la vérification des instances. Le terme de *caractérisation* est également employé pour dénoter, par extension, le réseau de contraintes vérifiées qui résulte du processus.

Dérivation de grammaire de propriétés sur corpus arboré

Étant donnée une grammaire hors-contexte implicite à un corpus arboré, la dérivation d'une grammaire GP requiert l'application de règles spécifiques à la sémantique de chaque type de propriété. Ces règles de dérivation que nous décrivons maintenant sont très largement inspirées de celles déjà décrites par Blache & Rauzy (2012).

Soit C une étiquette de nœud non-terminal, nous notons R_C l'ensemble des règles hors-contexte de partie gauche C, et définissons l'application RHS, qui à chaque C associe l'ensemble $\mathrm{RHS}(R_C)$ des parties droites pour C (une partie droite étant vue comme une liste d'étiquettes). Nous définissons également l'application label, qui associe un nœud x à son étiquette.

Règle 0 (Obligation – non encore implantée). La sémantique de l'Obligation sert principalement à l'identification des têtes de syntagme. Ainsi, en théorie, la propriété d'Obligation $C : \triangle H_C$ est spécifiée par l'ensemble H_C des disjonctions $\psi = \bigvee e$ des étiquettes distinctes e, tel qu'une étiquette e est présente dans toute règle $\mathrm{RHS}(R_C)$. Son implantation requiert la résolution d'un problème de couverture maximale d'ensembles réputé NP-dur.

Des travaux à venir implanteront un algorithme glouton d'approximation. Une autre option pourra consister à faire appel à une série d'heuristiques d'identification des têtes, sur le modèle de Dybro-Johansen (2004).

Règle 1 (Constituance). La propriété de Constituance $C : E_C?$ est spécifiée pour le syntagme C par l'ensemble E_C d'étiquettes uniques e, tel qu'il existe $r \in \mathrm{RHS}(R_C)$ avec $e \in r$.

$$E_C \equiv \{e, \exists x \exists r (r \in \mathrm{RHS}(R_C)) \wedge (x \in r) \wedge \mathrm{label}(x) = e\}$$

La figure 2 donne en exemple quelques propriétés de Constituance dérivées du corpus Sequoia (Candito & Seddah, 2012). Dans cet exemple et ceux qui suivent chaque propriété ré est présentée selon le patron `ETIQUETTE_SYNTAGME : liste_de_constituants`.

```
AdP: [DET, Srel, Ssub, NP, ADV, COORD, PP]
SENT: [NC, NP, ADV, VPpart, VN, VPinf, ADVWH, PP, AdP, I, Srel, Ssub, AP, Sint,
       NPP, COORD]
```

FIGURE 2 – Exemple de propriétés de Constituance dérivées du corpus Sequoia

Règle 2 (Unicité). La propriété d'Unicité $C : U_C!$ est spécifiée pour le syntagme C par l'ensemble U_C de toutes les étiquettes uniques qui ne co-occurent jamais avec elles-mêmes au sein de la même partie droite de règle.

$$\begin{aligned} U_C \equiv \{&e, \forall x \forall y \forall r \\ &r \in \mathrm{RHS}(R_C) \wedge (x \in r) \wedge (y \in r) \wedge ((\mathrm{label}(x) = \mathrm{label}(y)) \rightarrow (x = y)) \\ &\wedge \mathrm{label}(x) = e\} \end{aligned}$$

La figure 3 donne l'exemple de quelques propriétés d'Unicité dérivées de Sequoia.

```
COORD: [DET, AdP, CC, Srel, Ssub, AP, Sint, VPpart, VPinf, VN]
VPinf: [VPpart, CLO, VPinf, AdP, VINF, Sint]
PP: [PROREL, NC, P, ADJ, VPpart, VPinf, PRO, AdP, P+D, AP, Sint, P+PRO]
```

FIGURE 3 – Exemple de propriétés d'Unicité dérivées du corpus Sequoia

Règle 3 (Linéarité). L'ensemble des propriétés de Linéarité $C : a \prec b$ pour C est défini par l'ensemble L_C des paires ordonnées d'étiquettes (a, b) consistantes, où (a, b) est consistante ssi il existe une règle $r \in R_C$, telle que a et b co-occurent en partie droite de r, et il n'existe aucune règle $r' \in R_C$ telle que $(b, a) \in r'$. Nous notons i_x l'index du nœud x dans la liste en partie droite de règle.

$$\begin{aligned} L_C \equiv \{&(a,b), \forall x \forall y \forall r \exists x' \exists y' \neg \exists r' \\ &r \in \mathrm{RHS}(R_C) \wedge (x \in r) \wedge (y \in r) \wedge \mathrm{label}(x) = a \wedge \mathrm{label}(y) = b \wedge (i_x < i_y) \\ &\wedge r' \in \mathrm{RHS}(R_C) \wedge (x' \in r') \wedge (y' \in r') \wedge \mathrm{label}(x') = a \wedge \mathrm{label}(y') = b \\ &\wedge (i_{y'} < i_{x'})\} \end{aligned}$$

La figure 4 donne quelques exemples de propriétés de Linéarité dérivées de Sequoia.

```
VN: [[V, VPP], [VINF, AdP], [CLS, VPP], [V, PP], [CLS, ADV], [CLO, VS], [CLO, VINF], [PP, ADV],
    [CLO, VPP], [CLR, VPP], [VS, VINF], [VS, VPP], [CLS, VS], [PP, VPP], [V, COORD], [V, AdP],
    [VPR, VINF], [V, NP], [CLR, VS], [CLO, V], [VIMP, CLO], [VPR, VPP], [CLR, CLO], [CLO, VPR],
    [AdP, VPP], [CLO, AdP], [V, VINF], [CLR, VPR], [CLS, AdP], [NP, VPP]]
COORD: [[CC, VN], [CC, PP], [ADV, Ssub], [VPpart, NP], [Sint, PP], [CC, VPpart], [AdP, NP],
    [CC, VPinf], [CC, AP], [CC, Sint], [VN, Ssub], [CC, Srel], [CC, Ssub], [CC, AdP], [VN, AP],
    [CC, ADV], [ADV, PP], [VN, VPinf], [CC, DET], [AP, VPinf], [ADV, VPinf], [AP, PP], [CC, NP],
    [NP, Sint], [VN, VPpart]]
```

FIGURE 4 – Exemple de propriétés de Linéarité dérivées du corpus Sequoia

Règle 4 (Exigence). Rappelons que la sémantique de la propriété d'Exigence $C : x \Rightarrow y$ diffère de celle classique de l'implication, en ceci que contrairement à l'implication $(C : x \Rightarrow y) \not\equiv (\neg x \vee y)$. Donc l'ensemble Z_C des propriétés d'Exigence est spécifié par l'ensemble des co-occurrences au sein d'un même syntagme, moins les co-occurences pour lesquelles l'élément qui intervient en opérande gauche de la propriété peut apparaître dans une règle sans l'élément de l'opérande droit. Parmi l'ensemble des co-occurrences (a, b) pour C, nous retirons donc celles pour lesquelles il existe une règle $r \in \mathrm{RHS}(R_C)$ telle que $a \in r$ et $b \notin r$.

$$\begin{aligned} Z_C \equiv \{&(a,b), \forall x \forall r \exists y \\ &r \in \mathrm{RHS}(R_C) \wedge (x \in r) \wedge (y \in r) \wedge (x \neq y) \wedge ((\mathrm{label}(x) = a) \rightarrow ((\mathrm{label}(y) = b)))\} \end{aligned}$$

La règle 4 est seulement une approximation puisqu'elle ne capture aucun y disjonctif, comme l'autorise al sémantique de la propriété. Nous discutons les conséquences de cette limitation en §5 et envisageons des améliorations possibles. La figure 5 donne quelques exemples de propriétés d'Exigence dérivées de Sequoia.

Règle 5 (Exclusion). L'ensemble des propriétés d'Exclusion $C : x \not\Rightarrow y$ est spécifié par l'ensemble X_C des paires non-ordonnées d'étiquettes (a, b), telles que a et b ne co-occurrent jamais au sein de la même partie droite de règle.

$$\begin{aligned} X''_C \equiv \{&(a,b), \exists x \exists y \exists r \exists r' \\ &r \in \mathrm{RHS}(R_C) \wedge (x \in r) \wedge \mathrm{label}(x) = a \\ &r' \in \mathrm{RHS}(R_C) \wedge (y \in r') \wedge \mathrm{label}(y) = b\} \\ X'_C \equiv \{&(a,b), \exists x \exists y \exists r \\ &r \in \mathrm{RHS}(R_C) \wedge (x \in r) \wedge (y \in r) \wedge (x \neq y) \wedge \mathrm{label}(x) = a \wedge \mathrm{label}(y) = b\} \\ X_C \equiv &X''_C \backslash X'_C \end{aligned}$$

```
AdP: [[Ssub, ADV], [DET, PP], [NP, ADV], [Srel, ADV], [DET, ADV], [COORD, ADV], [PP, ADV]]
SENT: [[ADVWH, VN], [I, ADV], [Srel, VN], [I, AP], [I, VN], [ADVWH, PP]]
Srel: [[AP, NP], [AdP, AP], [ADV, VN], [PROREL, PP], [VPpart, NP], [AdP, NP], [AdP, VPpart],
    [PROREL, VN], [AdP, VN], [VPpart, VN], [PROREL, NP], [VPinf, VN], [AdP, ADV], [AP, VN],
    [Ssub, VN], [VPinf, NP], [COORD, VN]]
Ssub: [[VPpart, CS], [AdP, CS], [VPinf, VN], [Ssub, CS], [AdP, Sint], [VPinf, NP]]
AP: [[PP, ADJ], [VPinf, ADJ], [COORD, ADJ], [PREF, ADJ], [AdP, ADJ]]
NP: [[AdP, PP], [VPinf, NC], [VPinf, DET], [VN, PRO], [AdP, NC], [VPinf, PP], [DETWH, NC]]
```

FIGURE 5 – Exemple de propriétés d'Exigence dérivées du corpus Sequoia

La règle 5 peut être excessivement restrictive, dans la mesure où elle énumère la liste exhaustive des co-occurences interdites. Cependant, nous n'avons pas de meilleure solution à proposer pour le moment pour la dérivation de cette propriété. La figure 6 donne quelques exemples de propriétés d'Exclusion dérivées de Sequoia.

```
AdP: [[DET, Srel], [Ssub, COORD], [Srel, PP], [NP, COORD], [Srel, COORD], [DET, NP],
     [NP, PP], [Srel, Ssub], [DET, Ssub], [Ssub, PP], [DET, COORD], [COORD, PP], [Srel, NP]]
SENT: [[ADVWH, Ssub], [AdP, I], [Srel, Sint], [ADVWH, AP], [NC, ADVWH], [Srel, Ssub],
    [ADVWH, COORD], [NC, VPinf], [I, COORD], [NC, NPP], [VPpart, ADVWH], [NC, PP], [I, NPP],
    [AP, NPP], [NC, I], [PP, NPP], [ADVWH, NPP], [VN, NPP], [VPpart, AdP], [NPP, COORD],
    [NC, ADV], [NP, I], [AdP, AP], [ADVWH, AdP], [VPpart, I], [NC, NP], [ADVWH, Srel],
    [Srel, AP], [NC, VPpart], [ADV, NPP], [AdP, NPP], [Srel, NPP], [NC, Ssub], [Srel, COORD],
    [PP, I], [NC, COORD], [VPpart, NPP], [I, Srel], [VPinf, NPP], [AdP, COORD], [ADVWH, Sint],
    [I, Ssub], [NC, AP], [Ssub, NPP], [NC, Srel], [NP, ADVWH], [VPpart, Srel], [NC, Sint],
    [ADVWH, I], [Sint, NPP], [ADV, ADVWH], [NC, AdP], [NC, VN], [VPinf, Srel], [VPinf, I],
    [VPinf, ADVWH], [I, Sint]]
```

FIGURE 6 – Exemple de propriétés d'Exclusion dérivées du corpus Sequoia

5 Expérimentation : jugement d'analyses probables

Cette expérimentation vise à montrer qu'il est possible d'identifier partiellement les analyses agrammaticales générées par un analyseur probabiliste, en effectuant une vérification de modèle sur l'arbre généré.

Nous utilisons le Stanford Parser (Green *et al.*, 2011) (STP), et le corpus Sequoia (Candito & Seddah, 2012, ci-après CSP12). La partition du corpus en *développement* et *test* est identique à celle de CSP12. La grammaire GP utilisée pour ces expériences dérive du corpus d'entrainement du STP. Le tableau 2 résume les différentes valeurs obtenues lors des étapes d'extraction de la grammaire hors-contexte implicite, et de la dérivation de la grammaire GP. Un échantillon de la grammaire résultante est illustré par les figures 2 à 6. Après entraînement, le STP est utilisé pour parser le corpus de test.

Règles lexicales (POS-tags)	Règles syntagmatiques	TOTAL
5817	1409	7226

Constituance	Unicité	Linéarité	Exigence	Exclusion
165	108	321	99	678

TABLE 2 – Contenu de la grammaire GP dérivée

Les mesures PARSEVAL obtenues avec le programme `evalb` [10] sont reportées dans le tableau 3. Chaque analyse produite

10. `http://nlp.cs.nyu.edu/evalb/`, version du 2 novembre 2013.

Précision	Rappel	FMesure	Correspondance exacte	Nb phrases valides / Total
68.51	70.46	69.47	205 (19.65%)	1043 / 1043

TABLE 3 – Mesures PARSEVAL pour l'analyse du corpus de test de Sequoia par le STP

par le STP est ensuite caractérisée : son réseau de contraintes associé est créé pour la grammaire GP, et la satisfaction de chaque propriété du réseau est simultanément vérifié (processus de caractérisation). Le détail des caractérisations négatives des phrases jugées agrammaticales est rapporté dans le tableau 4. Sur un total de 1043 phrases vérifiées, 36 sont jugées agrammaticales. Elles représentent 3,45 % du total, et $36/838 \simeq 4.3\%$ des correspondances incomplètes au gold standard. Ce résultat doit être mis en perspective par rapport à différents éléments. Le premier est l'ambiguité syntaxique inhérente au langage naturel, qui peut expliquer qu'une analyse puisse être grammaticale bien que différente de celle du gold standard. Le deuxième tient au caractère incomplet de la grammaire GP utilisée pour l'expérimentation. Elle est incomplète pour plusieurs raisons : (a) la dérivation de la propriété d'Obligation, qui permet l'identification des têtes de syntagmes, n'est pas encore implantée ; (b) l'implantation de la propriété d'Exigence ne traduit pas la sémantique effective de manière satisfaisante ; (c) d'autres propriétés, telles que la Dépendance et l'Accord ne sont pas encore implantées non plus, du fait qu'elles requièrent l'introduction de structures de traits typés. Nous conjecturons que les développements à venir du processus de dérivation d'une grammaire GP devraient conduire à de meilleurs résultats.

Nous observons également — bien que nous n'ayons pas mesuré le phénomène — que différents patterns apparaissent qui informent sur la façon dont le STP choisit les analyses solutions. Le schéma d'annotation du corpus, notamment, semble influencer la dérivation de la grammaire par modèles à travers des étiquettes sur-spécifiées. Par exemple, le fait que les Noms Communs (NC) soient distingués des Noms Propres (NP) ne permet pas à la dérivation actuelle de conduire à une propriété qui spécifierait que la présence d'un déterminant dans un syntagme requiert la présence d'un nom (quelconque). Le problème peut probablement être résolu à l'aide, une fois encore, par l'introduction d'une structure de traits qui soit extraite du corpus. Néanmoins, même avec une telle structure le seul schéma d'annotation tel qu'utilisé dans le corpus Sequoia devrait rapidement montrer ses limites en termes de précision de l'information, et s'avérer cette fois sous-spécifié.

Un autre pattern récurrent est, semble-t-il, le choix peut-être un peu trop rapide par le parseur d'annotations morphologiques sous-optimales pour les mots de l'énoncé à analyser. Cette observation nous conduit à spéculer quant aux possibilités d'améliorer les performances d'un analyseur probabiliste tel que le STP. Dans la mesure où les meilleurs annotateurs morphologiques du moment affichent des exactitudes généralement supérieures à 97%, une hypothèse que nous formulons serait d'adopter une stratégie de reclassement (*reranking*) plus sophistiquée, où la meilleure annotation morphologique auraient un poids plus important dans le choix de la meilleure analyse syntaxique.

Enfin, il semble raisonnable de supposer qu'en intégrant la vérification de modèle dans le reclassement des n-meilleures analyses probables, il serait alors possible de préférer systématiquement les solutions jugées grammaticales aux solutions non-grammaticales, et améliorer ainsi les performances qualitatives de l'analyseur. Cette hypothèse reste cependant à vérifier.

6 Conclusion et perspectives

Nous venons de montrer qu'une représentation par modèles de la syntaxe du langage naturel peut s'allier avec succès à l'analyse syntaxique probabiliste afin de contribuer à résoudre des problèmes tels que le jugement de grammaticalité. L'absence de jugement de grammaticalité associé à un arbre syntaxique probable issu d'un analyseur probabiliste est, en effet, une carence préjudiciable dans de nombreux contextes applicatifs. La représentation par modèles associe un réseau de contraintes à la structure syntaxique. Ce réseau contient une information plus fine que la seule structure syntagmatique, puisqu'il intègre, pour chaque syntagme, l'ensemble des propriétés satisfaites et violées entre ses constituants. Ce réseau permet d'établir aisément un jugement exact de grammaticalité par vérification de modèle. L'étude expérimentale conduite sur les analyses générées par le Stanford Parser pour le corpus Sequoia a montré que près de 3,5 % des arbres générés (ou 4.3% des arbres ne correspondant pas à la référence) sont jugés agrammaticaux. Ce résultat permet de spéculer que la prise en compte du jugement de grammaticalité dans le classement des n solutions les plus probables par un parseur probabiliste devrait permettre une amélioration substantielle du résultat déterministe.

Num.		Règle de réécriture	Propriété	Instances violées
5	VN →	[ADV, V, AdP, VPP, VPP]	Exclusion	P- = [[ADV, AdP]]
18	SENT→	[NP, Sint, VN, VPinf, COORD]	Linearity	P- = [[VPinf, Sint], [COORD, Sint]]
41	SENT→	[NP, VN, PP, COORD, PP, VPpart]	Linearity	P- = [[VPpart, COORD]]
88	SENT→	[VPinf, VN, NP, VPinf, Sint]	Uniqueness	P- = [VPinf]
151	SENT→	[NP, VN, ADV, NP, Sint, VPinf]	Linearity	P- = [[VPinf, Sint]]
178	Srel →	[NP, VN, VPinf, PP]	Linearity	P- = [[PP, VPinf]]
	SENT→	[ADV, VN, VPinf, Srel]	Exclusion	P- = [[VPinf, Srel]]
214	SENT→	[ADV, NP, VN, NP, Sint, COORD]	Linearity	P- = [[COORD, Sint]]
264	SENT→	[PP, VN, AdP, NP, NP, PP]	Linearity	P- = [[AdP, PP]]
273	Srel →	[PP, VN, VPinf, Ssub]	Requirement	P- = [[VPinf, NP]]
	→	Exclusion	P- = [[Ssub, VPinf]]	
275	Sint →	[Ssub, NP, NP, NP, VN, AdP, NP]	Exclusion	P- = [[AdP, Ssub]]
278	Sint →	[Ssub, NP, VN, ADV, VPinf]	Linearity	P- = [[VPinf, Ssub]]
284	Sint →	[Ssub, NP, VN, VPinf]	Linearity	P- = [[VPinf, Ssub]]
322	Sint →	[Ssub, ADV, NP, VN, VPinf]	Linearity	P- = [[VPinf, Ssub]]
325	SENT→	[VN, ADV, NP, COORD, Ssub, VPpart]	Linearity	P- = [[VPpart, COORD]]
331	Sint →	[VPpart, NP, VN, PP, NP]	Linearity	P- = [[PP, VPpart]]
	Sint →	[VN, PP, Sint, VPpart]	Requirement	P- = [[Sint, NP], [VPpart, NP]]
	→	Exclusion	P- = [[VPpart, Sint]]	
337	Srel →	[PP, NP, VN, AP, PP, PP, Ssub]	Exclusion	P- = [[Ssub, AP]]
368	VPinf→	[VN, NP, VPpart]	Requirement	P- = [[VPpart, PP]]
418	SENT→	[ADVWH, VN, ADV, PP]	Exclusion	P- = [[ADV, ADVWH]]
419	VN →	[ADV, V, AdP, VPP]	Exclusion	P- = [[ADV, AdP]]
457	Sint →	[Ssub, NP, VN, VPinf]	Linearity	P- = [[VPinf, Ssub]]
465	Sint →	[Ssub, VN, VPinf]	Linearity	P- = [[VPinf, Ssub]]
469	SENT→	[PP, Sint, NP, NP, VN, VPinf, VPpart]	Linearity	P- = [[VPpart, Sint], [VPinf, Sint]]
481	SENT→	[VN, PP, Sint, AdP, PP]	Linearity	P- = [[AdP, PP]]
484	VPinf→	[VN, AP, Ssub]	Exclusion	P- = [[Ssub, AP]]
500	SENT→	[NP, Sint, NP, VN, ADV, NP, COORD, Ssub]	Linearity	P- = [[COORD, Sint]]
502	SENT→	[NP, Sint, VN, VPinf, ADV, Srel]	Linearity	P- = [[VPinf, Sint]]
	→	Exclusion	P- = [[VPinf, Srel], [Srel, Sint]]	
575	SENT→	[NP, Sint, VN, PP, VPpart]	Linearity	P- = [[VPpart, Sint]]
711	SENT→	[VN, ADV, NP, Sint, VPinf, Sint]	Linearity	P- = [[VPinf, Sint]]
744	Sint →	[VPpart, PP, VN, PP]	Linearity	P- = [[PP, VPpart]]
	→	Requirement	P- = [[VPpart, NP]]	
756	Ssub →	[CS, Sint, VN, NP, PP]	Exclusion	P- = [[Sint, VN], [NP, Sint], [Sint, PP]]
760	SENT→	[VPpart, NP, VN, PP, Sint, Sint, COORD]	Linearity	P- = [[COORD, Sint]]
779	Sint →	[VPpart, VN, PP]	Linearity	P- = [[PP, VPpart]]
	→	Requirement	P- = [[VPpart, NP]]	
799	SENT→	[NP, Sint, VN, ADV, VPinf]	Linearity	P- = [[VPinf, Sint]]
898	SENT→	[NP, VN, PP, NP, Sint, COORD]	Linearity	P- = [[COORD, Sint]]
1006	SENT→	[NP, Sint, NP, VPpart, VN, ADV, PP]	Linearity	P- = [[VPpart, Sint]]
1022	SENT→	[NP, VN, AP, NP, Srel]	Exclusion	P- = [[Srel, AP]]

TABLE 4 – Détail des jugements négatifs de grammaticalité

Références

BALFOURIER J.-M., BLACHE P. & RULLEN T. V. (2002). From Shallow to Deep Parsing Using Constraint Satisfaction. In *Proc. of the 6th Int'l Conference on Computational Linguistics (COLING 2002)*.

BENDER E. M., FLICKINGER D., OEPEN S., WALSH A. & BALDWIN T. (2004). Arboretum : Using a precision grammar for grammar checking in CALL. In *Proceedings of InSTIL/ICALL2004–NLP and Speech Technologies in Advanced Language Learning Systems–Venice*, volume 17, p.̃19.

BÈS G. & BLACHE P. (1999). Propriétés et analyse d'un langage. In *Proceedings of the 1999 Conference on Traitement Automatique du Langage Naturel (TALN'99)*.

BLACHE P. (2001). *Les Grammaires de Propriétés : des contraintes pour le traitement automatique des langues naturelles*. Hermès Sciences.

BLACHE P. & RAUZY S. (2012). Enrichissement du ftb : un treebank hybride constituants/propriétés.

CANDITO M. & SEDDAH D. (2012). Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *Actes de TALN'2012*, Grenoble, France.

CHARNIAK E. & JOHNSON M. (2005). Coarse-to-Fine n-Best Parsing and MaxEnt Discriminative Reranking. In *ACL*.

DUCHIER D., DAO T.-B.-H., PARMENTIER Y. & LESAINT W. (2010). Property grammar parsing seen as a constraint optimization problem. In *FG*, p. 82–96.

DUCHIER D., PROST J.-P. & DAO T.-B.-H. (2009). A model-theoretic framework for grammaticality judgements. In *Formal Grammar*, p. 17–30.

DYBRO-JOHANSEN A. (2004). *fExtraction automatique d'une grammaire d'arbres adjoints à partir du corpus arboré de Paris 7*. Dea de linguistique théorique et formelle, UFR Linguistique, Université Paris 7. sous la direction de M. Alexis Nasr, Dr.

FOSTER J., WAGNER J. & VAN GENABITH J. (2008). Adapting a wsj-trained parser to grammatically noisy text. In *ACL (Short Papers)*, p. 221–224.

GREEN S., DE MARNEFFE M.-C., BAUER J. & MANNING C. D. (2011). Multiword Expression Identification with Tree Substitution Grammars : A Parsing tour de force with French. In *EMNLP 2011*.

MARUYAMA H. (1990). Structural Disambiguation with Constraint Propagation. In *Proceedings 28th Annual Meeting of the ACL*, p. 31–38, Pittburgh, PA.

MUTTON A., DRAS M., WAN S. & DALE R. (2007). GLEU : Automatic Evaluation of Sentence-Level Fluency. In *ACL*.

PRINCE A. & SMOLENSKY P. (1993). *Optimality Theory : Constraint Interaction in Generatire Grammar*. Rapport interne, TR-2, Rutgers University Cognitive Science Center, New Brunswick, NJ.

PROST J.-P. (2008). *Modelling Syntactic Gradience with Loose Constraint-based Parsing*. PhD thesis, Macquarie University, Sydney, Australia, and Université de Provence, Aix-en-Provence, France (cotutelle).

PULLUM G. & SCHOLZ B. (2001). On the Distinction Between Model-Theoretic and Generative-Enumerative Syntactic Frameworks. In P. DE GROOTE, G. MORRILL & C. RÉTORÉ, Eds., *Logical Aspects of Computational Linguistics : 4th International Conference*, number 2099 in Lecture Notes in Artificial Intelligence, p. 17–43, Berlin : Springer Verlag.

TETREAULT J. R. & CHODOROW M. (2008). The ups and downs of preposition error detection in esl writing. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, p. 865–872 : Association for Computational Linguistics.

WAGNER J. (2012). *Detecting Grammatical Errors with Treebank-Induced, Probabilistic Parsers*. PhD thesis, Dublin City University, Dublin, Ireland.

WAN S., DRAS M., DALE R. & PARIS C. (2005). Towards statistical paraphrase generation : preliminary evaluations of grammaticality. In *Proceedings of The 3rd International Workshop on Paraphrasing (IWP2005)*, p. 88–95.

WONG S.-M. J. & DRAS M. (2011). Exploiting Parse Structures for Native Language Identification. In *EMNLP*, p. 1600–1610.

ZWARTS S. & DRAS M. (2008). Choosing the right translation : A syntactically informed classification approach. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, p. 1153–1160 : Association for Computational Linguistics.

Annotation sémantique et validation terminologique en texte intégral en SHS

Résumé. Nos travaux se focalisent sur la validation d'occurrences de candidats termes en contexte. Les contextes d'occurrences proviennent d'articles scientifiques des sciences du langage issus du corpus SCIENTEXT[1]. Les candidats termes sont identifiés par l'extracteur automatique de termes de la plate-forme TTC-TermSuite et sont ensuite projetés dans les textes. La problématique générale de cet article est d'étudier dans quelle mesure les contextes sont à même de fournir des critères linguistiques pertinents pour valider ou rejeter chaque occurrence de candidat terme selon qu'elle relève d'un usage terminologique en sciences du langage ou non (langue générale, transdisciplinaire, autre domaine scientifique). Pour répondre à cette question, nous comparons deux méthodes d'exploitation (l'une inspirée de la textométrie et l'autre de Lesk) avec des contextes d'occurrences du même corpus annotés manuellement et mesurons si une annotation sémantique des contextes améliore l'exactitude des choix réalisés automatiquement.

Abstract. Our work is in the field of the validation of term candidates occurrences in context. The textual data used in this article comes from the freely available corpus SCIENTEXT. The term candidates are computed by the platform TTC-TermSuite and their occurrences are projected in the texts. The main issue of this article is to examine how contexts are able to provide relevant linguistic criteria to validate or reject each occurrence of term candidates according to the distinction between a terminological and a non terminological use (general language, transdisciplinary use, use coming from another science). To answer this question, we compare two methods (a textometric one and another inspired from Lesk) with the manual annotation of the same corpus and we evaluate if a semantic annotation of contexts improves the accuracy of the choices made automatically.

Mots-clés : Annotation sémantique - extraction et désambiguïsation terminologique - textométrie - texte intégral
Keywords: Semantic Annotation - Terminological Extraction and Disambiguation - Textual Metrics (Specificity) - Full Text

1 Introduction

Nos travaux se situent dans le champ de l'extraction terminologique à partir de textes intégraux dans le domaine des SHS et plus précisément celui des sciences du langage. A la suite de (Daille 1994), (Toussaint et *al.* 1998), (Bourigault et Slozdian 1999), (Bourigault et. *al* 2001) parmi beaucoup d'autres, nous privilégions une approche allant du texte (réalisations linguistiques de termes dans les textes) aux termes (objets conceptuels). Accéder aux réalisations linguistiques des termes dans les textes suppose de les reconnaître comme telles. Parmi les travaux qui abordent cette problématique, une première partie s'appuie sur l'utilisation d'extracteurs automatiques de candidats termes qui sont ensuite validés par des experts des domaines de spécialités concernés : *Acabit* (Daille 1994 ; 2003), *Yatea* (Aubin et Hamon 2006), *TermoStat* (Drouin 2003) ou encore la plate-forme *TTC-TermSuite* (Daille et *al.*, 2011). D'autres travaux, qui peuvent être connexes, s'intéressent à la validation, l'extraction de candidats termes ou de relations entre termes dans les textes, en mettant en œuvre différentes exploitations des textes dans une approche distributionnelle. Les travaux de Daille (2003), Toussaint et *al.* (1998), Namer et Zweigenbaum (2004) ou M-C L'Homme (2004a) s'appuient sur des connaissances relevant de la morphologie dérivationnelle ou constructionnelle. Les travaux de Baneyx et *al.* (2005), Jacques et Aussenac-Gilles (2006), Aussenac-Gilles et Condamines (2009), Manser (2012), Périnet et Hamon (2013) détectent et exploitent des patrons lexico-syntaxiques pour l'identification de relations entre (réalisations linguistiques de) termes. Enfin, les travaux de Grabar et Zweigenbaum (2004), Claveau et M-C L'Homme (2005), Poibeau (2005) ou Condamines et Péry-Woodley (2007) reposent sur l'utilisation de structures sémantiques, textuelles ou discursives.

Notre objectif est d'analyser des contextes d'occurrences de candidats termes qui sont sémantiquement enrichis afin de sélectionner automatiquement les occurrences de candidats qui relèvent d'un usage terminologique et de rejeter les autres. Autrement dit, nous procédons à un type particulier de désambiguïsation sémantique que nous appellerons « désambiguïsation terminologique ». En effet, on peut constater, à la suite de M.C. L'Homme (2004b), que même si le terme en tant qu'étiquette de concept, pour une terminologie donnée et une application définie, n'est pas ambigu, ses

[1] Scientext permet d'accéder à un outil d'interrogation pour l'ensemble de la base de textes du projet. Une partie de la base est accessible est sous licence Creative Common : http://scientext.msh-alpes.fr/scientext-site/?article8 [pages consultées le 12/02/2014]

réalisations linguistiques peuvent l'être. Ceci est vrai en particulier lorsque les « termes » sont des « candidats termes » extraits automatiquement par une plate-forme d'extraction terminologique. Ce phénomène pourrait aussi se manifester lorsque les termes dont on observe les occurrences en texte intégral appartiennent à des thésaurus ou des référentiels terminologiques.

– ambiguïté avec le lexique ou la phraséologie transdisciplinaire : *argument, corpus, définition, énoncé, exemple, objet, référence*
  [+term] *En chemin, nous avons souligné la grande flexibilité des SN définis pluriel, qui en fait le lieu possible d'une négociation de la* ***référence*** *et de la désignation* (Figures et référence plurielle en corpus journalistique - Lecolle M. (2000). Cahiers de grammaire (25))
  [-term] *Les auteurs font* ***référence*** *à [...]* (Validation d'une méthodologie pour l'étude des marqueurs de la segmentation dans un grand corpus de texte - Piérard S. et Begsten Y. (2007). TAL(47/2))

– ambiguïté avec un autre domaine de spécialité : *patient*
  [+term] *[...] ou plus rarement, à des rôles argumentaux (agent,* ***patient****, objet,...) [...]* (Les relations sémantiques : du linguistique au formel - Aussenac-Gilles N. et Séguéla P. (2000). Cahiers de grammaire (25))
  L[-term] *es* ***patients*** *cérébrolésés [...]* (Nouveaux habits de la lexicographie spécialisée : Intégration de la métaphorique dans le dictionnaire du football - Leroyer P. et Moller B. (2004). EURALEX)

– ambiguïté avec un emploi lexical ou phraséologique de langue générale : *argument, définition, énoncé, expression, objet*
  [+term] *[...] les* ***expressions*** *du type le jour suivant.* (Validation d'une méthodologie pour l'étude des marqueurs de la segmentation dans un grand corpus de texte - Piérard S. et Begsten Y. (2007). TAL(47/2))
  [-term] ***L'expression*** *de telle ou telle relation [...]* (Variabilité des outils de TAL et genre textuel : cas des patrons lexico-syntaxiques - Jacques M.-P. et Aussenac-Gilles N. (2006). TAL (47))

Comme le montrent ces quelques exemples de réalisations linguistiques de candidats termes, c'est le contexte au sens large qui nous permet de sélectionner les occurrences relevant d'un emploi terminologique. Deux interrogations apparaissent alors : en ce qui concerne les différentes manières d'exploiter les contextes d'occurrences des candidats et la sélection des informations à exploiter dans ces contextes. Dans cet article, nous développons une approche qui s'appuie, d'une part, sur une analyse statistique des contextes d'occurrences, analyse fondée sur le calcul de spécificité (Lafon 1980), et d'autre part, sur des contextes annotés sémantiquement à l'aide de traits sémantiques ou quasi-sèmes qui sont extraits de deux ressources lexicales, le TLFi et WiktionnaireX. L'approche présentée est ensuite comparée à l'algorithme de Lesk (1986).

2 Données et méthodes

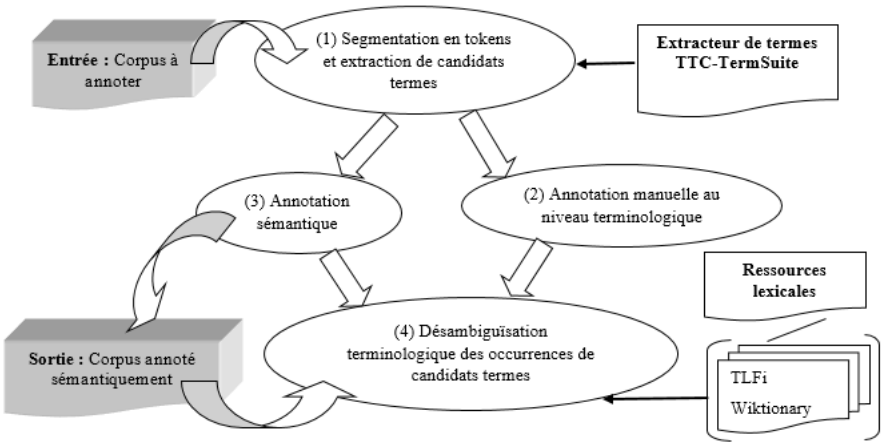


Figure 1: Méthodologie générale de l'expérience

La plate-forme TTC-TermSuite[2] segmente le corpus de travail en tokens, l'annote avec TreeTagger puis en extrait une terminologie, c'est-à-dire une liste de candidats termes qui est projetée dans les textes. Le corpus enrichi en candidats

[2] La plate-forme TTC-TermSuite est librement utilisable et open source. Elle est accessible sous licence Apache 2.0 https://code.google.com/p/ttc-project/downloads/detail?name=ttc-term-suite-1.4.jar [page consultée le 12/02/2014]

termes subit ensuite deux traitements parallèles. D'un côté, il est pris en charge dans une plate-forme d'annotation manuelle au sein de laquelle un annotateur linguiste expert désambiguïse manuellement les occurrences de candidats termes en les validant ou les rejetant d'un point de vue terminologique. A l'issue de l'annotation, les occurrences désambiguïsées de candidats termes sont considérées comme des occurrences de termes. Cet enrichissement fournit une version du corpus qui sert de corpus de référence. D'un autre côté, le corpus enrichi automatiquement en candidats termes est pris en charge par le module d'annotation sémantique. Le nouveau corpus obtenu, enrichi en candidats termes et en annotations sémantiques, est ensuite pris en charge par le module de désambiguïsation terminologique dont la tâche est de sélectionner pour chaque candidat ses occurrences terminologiques et de rejeter ses occurrences non terminologiques. Enfin, nous évaluons les performances de la désambiguïsation terminologique réalisée automatiquement en la comparant avec celle effectuée manuellement dans la plate-forme d'annotation manuelle.

2.1 Corpus de travail, enrichissement terminologique et corpus de référence

Le corpus de travail rassemble 62 articles appartenant au domaine scientifique des sciences du langage, extraits de la base Scientext. Ce corpus, au format xml-tei, comporte 397 695 occurrences. L'ensemble des textes se répartit en 47 articles de conférences (75,81% des documents et 57,06% des occurrences), et 15 articles de revues (24,19% des documents et 42,94% des occurrences). Ainsi, en nombre d'occurrences, le corpus utilisé est assez équilibré entre conférences et revues[3]. Le corpus de travail est traité par l'extracteur automatique de termes TTC-TermSuite afin d'obtenir une liste de candidats termes. Quatre paramètres définissent le filtrage de la liste des candidats termes en sortie : (1) regroupement par variantes flexionnelles et syntaxiques (la distance choisie est le *Log-likelihood ratio*) ; (2) seuil minimal de fréquence fixé à 5 ; (3) sélection des 7 500 (maximum) premiers candidats termes triés par spécificité décroissante[4] ; (4) ne sont considérés que les candidats nominaux et adjectivaux, simples et complexes. Un module de traitement interne projette ensuite les candidats termes dans les textes et les encapsule dans des balises XML tout en gérant les chevauchements tels que ceux que l'on trouve dans *graphes de dépendances sémantiques* entre *graphes de dépendances* et *dépendances sémantiques*. Le module de projection crée une nouvelle version du corpus enrichie en candidats termes.

Le corpus de référence est constitué d'une sélection de 52 documents extraits du corpus de travail enrichi en occurrences de candidats termes qui ont été évaluées manuellement du point de vue de leur caractère terminologique en contexte. Cette tâche a été réalisée au sein d'une interface d'annotation librement consultable[5] : les occurrences de candidats termes sont bornées par des crochets et leur empan est matérialisé par un surlignage dynamique. Une puce dont la couleur varie entre le vert et le rouge, cliquable dynamiquement représente les choix finaux de l'annotateur (le vert correspond à une validation, le rouge à un rejet). A l'issue de l'annotation manuelle, les évaluations sont stockées et décomptées. Parmi les 50 993 occurrences de candidats termes, correspondant à 4 431 candidats termes différents, 14 544 occurrences sont validées, soit 28,52 %. Le décompte des choix permet une classification des candidats sur deux échelles qui seront utilisées pour caractériser le jeu de test des expériences (section 3) :

- une échelle d'ambiguïté allant de 0 à 50 : nous avons opté pour cet intervalle de [0,50] suite au ratio (exprimé en pourcentage) « nombre d'occurrences validées / nombre total d'occurrences ». Plus il se rapproche de 50, plus le candidat est jugé ambigu ; pour les termes qui ont un ratio supérieur à 50, nous avons opté pour le complémentaire du ratio (100 - ratio).
- une échelle représentant la tendance terminologique du candidat (0-100): plus le ratio « nombre d'occurrences validées / nombre total d'occurrences » se rapproche de 100, plus le candidat a une tendance terminologique forte et inversement si ce ratio se rapproche de 0.

2.2 Enrichissement sémantique des données

Habituellement, lorsqu'on parle d'annotation et en particulier d'annotation sémantique, l'objectif est d'associer « une interprétation stabilisée » aux données brutes (Habert 2005). L'annotation sémantique dans cette perspective suppose la désambiguïsation. Dans l'expérience que nous menons, nous faisons le choix de procéder à un enrichissement sémantique ambiguë qui ne privilégie aucun des sens fournis par les ressources utilisées pour l'annotation. Nous

[3] Les conférences représentées sont le Cédil (Colloque international des Étudiants chercheurs en Didactique des Langues et en Linguistique), Euralex (Conférence de European Association of Lexicographie) et le colloque EID (Émotions, Interactions, Développements). Les revues sont TAL (Traitement automatique des langues), Les cahiers de grammaire et la revue LiDil (Revue de Linguistique et de Didactique des langues).

[4] Ce seuil de 7 500 a été déterminé de manière empirique par comparaison avec la distribution des types de structures de candidats complexes que l'on obtient sans filtrage.

[5] La page https://apps.atilf.fr/smarties/ [page consultée le 12/02/2014] permet un accès public en consultation. Le guide d'annotation, disponible sur le site, permet de comprendre comment est effectuée l'évaluation des candidats termes depuis leur état initial vers l'état de leur validation terminologique.

plaçant dans le cadre de la sémantique interprétative, l'une des hypothèses majeures sur laquelle nous nous appuyons est que le sens d'une unité lexicale est en grande partie co-construit par son contexte d'usage ou totalement construit pour le sens émergeant des usages[6]. C'est donc dans un deuxième temps (section 2.3 ci-dessous) que nous analysons l'information sémantique ajoutée en vue d'un type particulier de désambiguïsation sémantique, à savoir la désambiguïsation terminologique des occurrences de candidats termes apparaissant dans les contextes enrichis sémantiquement. Cette méthodologie permet de mesurer le caractère opératoire ou non du type d'annotation sémantique que nous appliquons lors du processus de désambiguïsation terminologique.

Comme nous l'avons précisé dans l'introduction (section 1), les informations sémantiques ajoutées aux unités lexicales sont des traits sémantiques ou quasi-sèmes. À la suite de Valette et *al.* (2006), nous faisons l'hypothèse que les mots pleins de toutes les définitions d'un mot vedette d'un dictionnaire représentent des traits sémantiques bruts associés à tous les sens possibles de ce mot vedette. Les traits sémantiques sont extraits sous forme lemmatisée et catégorisée en parties du discours *(adv*, adverbes ; *adj*, adjectifs ; *subst*, noms ; *v*, verbes). Ainsi, le nom *locuteur*, qui a pour définition dans le TLFi *Personne qui parle, qui produit des énoncés* sera représenté par l'ensemble de traits sémantiques {"*personne":subst, "parler":v, "produire":v, "énoncé":subst*}. Lorsque le mot vedette correspondant à une unité lexicale du texte à annoter a plusieurs définitions dans le dictionnaire, celles-ci sont ajoutées les unes aux autres. Enfin, lorsqu'un trait sémantique apparaît plusieurs fois, il est répété afin de tenir compte de toutes ses occurrences. C'est le cas du trait *note* dans l'une des définitions du verbe *annoter* dans le TLFi *"pourvoir un texte de notes" "mettre des notes en marge de ..."* : {*"pourvoir":v, "texte":subst, "note":subst, "mettre":v, "note":subst, "marge":subst*}.

La première ressource que nous utilisons est le TLFi car il s'agit d'une ressource libre sur signature d'une convention, à large couverture, comportant des définitions lemmatisées et catégorisées et disposant d'une structuration XML permettant l'extraction de ces définitions. Cependant, cette ressource n'est actuellement plus mise à jour et il est nécessaire de la compléter. La mise en ligne du WiktionnaireX par F. Sajous[7] permet d'étendre la couverture de la ressource initiale de manière significative. Nous avons sélectionné dans le WiktionnaireX (structuré en XML) les entrées de formes lemmatisées conformes au format des documents à annoter sémantiquement (les textes sont lemmatisés et annotés en parties du discours par TreeTagger via l'extracteur de candidats termes TTC-TermSuite). Pour catégoriser les traits sémantiques des définitions du WiktionnaireX, nous avons projeté les définitions catégorisées du TFLi. Nous obtenons ainsi deux ressources de traits sémantiques.

Pour choisir le mode d'annotation mis en œuvre dans les expériences (section 3), nous avons cherché à maximiser la couverture de l'annotation sémantique en utilisant les deux ressources de traits sémantiques (celle issue du TLFi prioritairement et celle issue du WiktionnaireX pour compléter). Les six types d'annotations qui ont été définis et dont le taux de couverture est représenté ci-dessous (figure 2) ont été définis en fonction de deux paramètres :

1. la ressource de traits sémantiques utilisée (TLFi exclusivement ; WiktionnaireX exclusivement ; TLFi complété par le WiktionnaireX) ;
2. les critères d'identification des entrées lexicales utilisées pour l'annotation des unités lexicales dans le textes
   - V1 = double correspondance établie à la fois sur la forme lemmatisée et la catégorie grammaticale
   - V2 = correspondance simple établie uniquement sur la forme lemmatisée

[6] On peut citer l'exemple bien connu aujourd'hui de l'usage néologique du nom *caviar* dans le discours journalistique principalement sportif. Dans ce genre textuel bien précis et pour le football en particulier, le nom *caviar* désigne une très belle passe comme l'atteste cet exemple repris de Rastier et Valette (2009 : 14) : *Confirmation ici, d'un centre précis, [David Beckham] trouvait la tête de Frank Lampard qui n'avait plus qu'à régler la mire pour ouvrir la marque et transformer ce* ***caviar*** *en but.* (Site sports.fr, 14.06.2004) .

[7] Cette ressource libre et open source est accessible à http://redac.univ-tlse2.fr/lexiques/wiktionaryx.html [page consultée le 04/02/2014]. Elle a notamment été utilisée dans (Sajous et *al.* 2013).

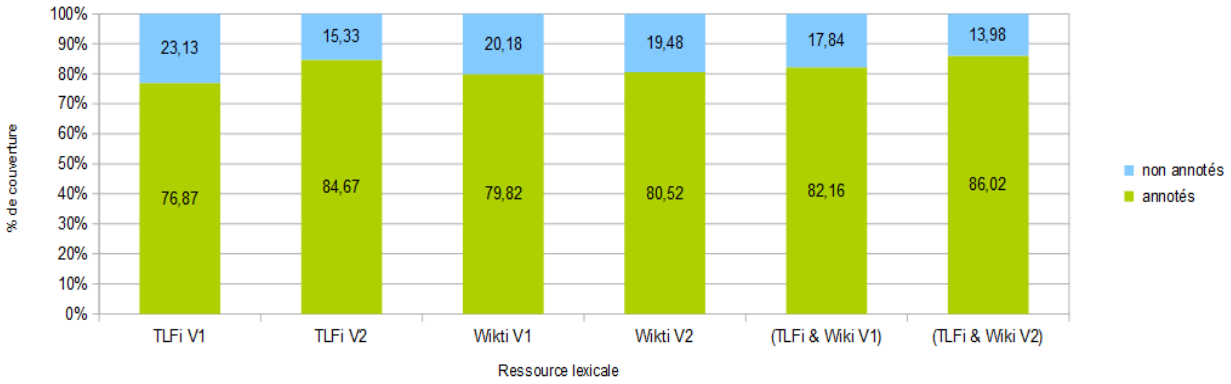


Figure 2: Couverture d'annotation de l'ensemble des mots du corpus Scientext à l'aide de différentes ressources lexicales

L'annotation (TLFi & WIKI V2) atteint une couverture de 86,02 %, elle représente le ratio entre le nombre de mots pleins de toute catégorie annotés par rapport au nombre total de mots pleins.
Pour conclure sur ce point, le tableau (1) ci-dessous détaille les taux de couverture obtenus par catégorie d'unité lexicale[8] et par type d'annotation. Là encore, le type (TLFi & Wiki V2) semble le plus satisfaisant et c'est donc ce type d'annotation qui a été choisi pour annoter le corpus de travail en traits sémantiques.

	Noms (%)		Verbes (%)		Adjectifs (%)		Adverbes (%)	
	Tokens	Types	Tokens	Types	Tokens	Types	Tokens	Types
TLFi V1	81,88	51,28	86,01	66,37	70,2	52,6	34,73	72,82
TLFi V2	82,31	52,21	95,2	67,53	75,45	55,24	85,33	91,02
Wiki V1	80,4	52,2	86,27	68,37	66,5	51,72	85,43	90,05
Wiki V2	80,74	53,27	86,43	69,08	68,25	54,3	87,64	94,17
TLFi & Wiki V1	83,27	54,37	86,35	68,73	71,77	55,48	85,43	90,05
TLFi & Wiki V2	**83,52**	**55,13**	**95,64**	**70,28**	**76,43**	**57,7**	**91,12**	**97,09**

Tableau 1: Résultat global d'annotation (couverture) du corpus Scientext à l'aide de différentes ressources lexicales

2.3 Désambiguïsation terminologique des occurrences de candidats termes

Par rapport aux champs d'application et aux difficultés relevés dans la problématique de la désambiguïsation sémantique en général (Navigli 2009), (Navigli et Lapata 2010), (Schwab et al. 2013), réaliser une désambiguïsation terminologique pourrait nous placer dans une configuration plus favorable : (1) seuls les candidats termes sont désambiguïsés et non l'ensemble des unités lexicales des textes ; (2) les « sens » entre lesquels arbitrer forment un ensemble borné et constant (usage terminologique vs. usage non terminologique).

Le type de désambiguïsation mis en œuvre appartient au champ des méthodes probabilistes supervisées. Suite à des travaux antérieurs qui ont donné des résultats encourageants (Camacho-Collados et *al.* 2014), nous utilisons une méthode basée sur la spécificité lexicale (Lafon 1980) dans laquelle l'indice de spécificité de chaque mot du corpus représente le sur-emploi ou sous-emploi des mots dans le sous-corpus par rapport au corpus de référence. Cette méthode est qualifiée de *méthode de désambiguïsation textométrique*. Le calcul de spécificité est utilisé pour établir, à partir du corpus de référence, des profils caractéristiques des deux types d'usage que l'on cherche à différencier. D'autres méthodes font usage exclusivement de ressources lexicales (sans utiliser un corpus de référence) et, plus spécifiquement de dictionnaires. On trouve parmi ces méthodes l'algorithme de Lesk (1986). Dans cette approche, un score est calculé pour la définition terminologique et un autre score pour les définitions relevant de la langue générale. Chaque score est déterminé à partir de l'intersection des mots d'une définition (terminologique ou non) avec les mots du contexte proche du candidat terme à désambiguïser. Étant donné la simplicité de l'algorithme, les résultats obtenus par Lesk sont intéressants en ce qu'ils permettent une comparaison moyennant une adaptation marginale. L'adaptation

[8] Les résultats entre types et tokens s'inversent pour les adverbes parce que les adverbes non annotés ont une fréquence d'apparition plus importante que ceux qui sont annotés. Par exemple, les séquences « a priori, ne, est-à-dire » apparaissent plusieurs fois et n'ont pas d'entrée dans le TLFi ou le le WiktionnaireX. En utilisant l'annotation (TLFi & Wiki V2), nous sommes parvenus à annoter 400 types parmi 412, alors que seulement 12899 tokens sont annotés parmi 14156.

porte sur le contexte considéré autour du mot à désambiguïser : au lieu du texte complet, nous prenons en compte le paragraphe. En effet, un texte peut contenir plusieurs occurrences d'un même candidat terme avec un comportement différent du point de vue terminologique, ce qui est beaucoup moins vrai du paragraphe.

2.3.1 Désambiguïsation textométrique

Cette première méthode de désambiguïsation, qui s'appuie sur les travaux antérieurs de Camacho Collados et al. (2014), est réalisée en deux temps : (1) établissement des profils caractérisant les usages à différencier ; (2) utilisation de ces profils pour décider, contexte par contexte, si l'occurrence analysée est terminologique ou non. Les profils terminologiques vs. non terminologiques sont établis à partir du corpus de référence en rassemblant pour chaque candidat dans deux ensembles disjoints les contextes où ce candidat relève d'un usage terminologique et les contextes où il relève d'un usage non terminologique. Chacun de ces ensembles constitue un sous-corpus par rapport au corpus de référence. Pour chaque candidat, on dispose donc d'un sous-corpus terminologique (SC_{on}) et d'un sous-corpus non terminologique (SC_{off}). Dans cette configuration, en appliquant à chaque sous-corpus (SC) un algorithme de calcul du taux de spécificité dont les résultats ont été vérifiés par comparaison avec ceux obtenus à l'aide du logiciel textométrique TXM (Heiden 2010), nous produisons une liste d'éléments supposés caractéristiques d'un usage terminologique (appelée LS_{on}) et une liste d'éléments supposés caractéristiques d'un usage non terminologique (appelée LS_{off}). Tous les éléments présents dans chacune de ces listes (LS) sont représentés avec leur taux de spécificité[9]. Pour créer les profils de traits sémantiques (LST), on répète le processus en remplaçant chaque mot plein par ses traits sémantiques. Dans le cas des contextes annotés en traits sémantiques, les profils établis sont désignés par LST_{on} et LST_{off} et ils sont construits à partir des sous-corpus disjoints annotés en traits sémantiques appelés respectivement SCT_{on} et SCT_{off}. Le tableau (2) donne une sélection des profils calculés pour le candidat *patient*.

LS_{on}		LST_{on}		LS_{off}		LST_{off}	
agent	9,51	*lar*	16,61	**douleur**	48,48	*vérifier*	123,20
planter	9,31	*eupatoriées*	10,33	**cérébrolésés**	30,06	*examen*	78,39
planteur	5,91	*dot*	9,36	**contrôle**	26,47	*pénible*	59,05
adventice	5,43	*enfoncer*	8,59	**récit**	20,68	*plaindre*	38,78
schème	4,73	*debout*	8,19	**trouble**	19,18	*poinçonner*	26,72
médecin	4,73	*primitif*	8,12	**maladie**	17,43	*platiner*	26,72
sen|sens	4,55	*actionnaire*	8,09	**adolescent**	14,48	*fisc*	26,72
primitif	4,17	*angleterre*	7,41	**frontal**	13,64	*physique*	26,71
conséquent	3,07	*agrément*	5,81	**dardier**	12,91	*examiner*	24,89
...		...		...		...	
patient (peu ambigu, peu terminologique)							

LS_{on}		LST_{on}		LS_{off}		LST_{off}	
temporel	19,09	*marquer*	35,72	**talmy**	12,98	*température*	40,43
chance	13,85	*grille*	34,54	**trajectoire**	10,67	*chaleur*	33,40
marqueur	12,39	*communiquer*	30,38	**typologie**	10,24	*place*	18,92
thème	11,99	*impact*	28,49	**cri**	10,08	*sensiblement*	18,67
bac	11,46	*choeur*	28,48	**slobin**	10,02	*typologie*	18,29
rupture	10,77	*balustrade*	24,47	**tunisien**	9,25	*sud*	13,28
thématique	9,73	*inscrire*	23,56	**interjection**	8,92	*sibérie*	12,48
émotionnel	9,11	*but*	22,55	**encoder**	8,87	*cordialité*	11,10
positif	8,81	*point*	20,34	**click**	8,59	*peur*	10,98
...		...		...		...	
expression (très ambigu)							

Tableau 2 : Profils caractéristiques des contextes terminologiques ou non terminologique extraits des contextes d'occurrences lexicaux ou enrichis en traits sémantiques[10]

[9] Le taux de spécificité de Lafon (1980) est calculé en comparant sur une distribution observée dans un corpus de référence et une distribution théorique d'un échantillon définie selon une loi hypergéométrique, : cette fréquence théorique de chaque mot est proportionnelle à la fréquence de ce mot dans le corpus. Le signe du taux de spécificité d'un élément est positif si sa fréquence observée est supérieure à sa fréquence théorique. La valeur d'un taux de spécificité se déduit de la probabilité d'obtenir la fréquence observée. (Reutenauer 2012 : chapII.2) donne une description détaillée de cet indice et le compare avec d'autres indices statistiques.

[10] Les éléments représentés dans les profils donnés ont fait l'objet d'une sélection à des fins de lisibilité : les 9 éléments indiqués sont les plus spécifiques de chaque profil et n'appartiennent pas à l'intersection des profils On x Off. Dans le calcul actuel, cette sélection n'est pas encore mise en œuvre et la différenciation des éléments communs s'appuie sur les taux de spécificité qui sont toujours très différents, par exemple *subir* avec le candidat *patient* qui a un taux de 25 en *ON* et un taux limite de 1000 en *OFF*.

Bien que le calcul des profils suive la même méthode que ceux-ci soient établis à partir des contextes lexicaux ou à partir des contextes enrichis en traits sémantiques, il existe deux différences. Les profils "lexicaux" sont constitués d'unités lexicales spécifiques des contextes et le seuil minimal du taux de spécificité est fixé à 1,5. Les profils "sémantiques" sont constitués de traits sémantiques et le seuil minimal du taux de spécificité est fixé à 2. Grâce à la multiplication des données à traiter suite à l'annotation en traits sémantiques (multiplication moyenne par 10), nous pouvons être plus sélectifs dans le choix des traits sémantiques[11].

La désambiguïsation de chaque occurrence des candidats est réalisée en comparant chaque contexte avec les profils terminologiques et non terminologiques. Les contextes lexicaux sont comparés avec les profils lexicaux LS_{on} et LS_{off} tandis que les contextes annotés sémantiquement sont comparés avec les profils sémantiques LST_{on} et LST_{off} . La comparaison est identique dans les deux cas. On détermine les éléments communs entre le contexte de l'occurrence à désambiguïser et les deux profils adéquats. Les taux de spécificité des éléments communs sont additionnés respectivement pour chaque profil, terminologique vs. non terminologique, ce qui permet d'obtenir deux scores : $score_{on}$ et $score_{off}$. L'occurrence est jugée terminologique si $score_{on}$ est supérieur à $score_{off}$ et elle est jugée non terminologique dans le cas contraire. Dans les résultats (section 3), la désambiguïsation textométrique sur les contextes lexicaux est désignée par l'abréviation *SpecLex* et celle qui est appliquée sur les contextes annotés en traits sémantiques est désignée par l'abréviation *SpecTraits*.

Le dernier point méthodologique à préciser est que les profils, lexicaux et sémantiques, sont systématiquement recalculés en éliminant le contexte de l'occurrence à désambiguïser. Ceci a pour but d'éviter de fausser les résultats de l'évaluation et d'éviter toute circularité.

2.3.2 Algorithme de Lesk

Cet algorithme (Lesk, 1986) vise à associer la bonne définition à chaque unité lexicale en contexte. Pour appliquer cet algorithme dans notre expérience, nous avons construit des définitions terminologiques pour les candidats termes en nous inspirant des usuels spécialisés en sciences du langage. Plusieurs ressources sont utilisées pour l'écriture des défintions terminologiques (Dubois J. et al. 2012 ; Ducrot O. et Schaeffer J-M. 1999 ; Neveu F. 2004). L'adaptation de l'algorithme de Lesk a consisté à choisir entre la définition terminologique d'un candidat et les définitions de ce candidat lorsqu'il est utilisé de manière non terminologique en sciences du langage. Ces défintitions sont prises dans les ressources de langue générale utilisées, le TLFi et WiktionnaireX[12]. Pour réaliser cette expérience, l'algorithme mesure l'intersection entre le contexte de chaque occurrence du candidat terme et les définitions candidates (la définition terminologique et les définitions de langue générale). Lorsque l'algorithme est appliqué aux contextes annotés en traits sémantiques (méthode appelée *LeskTraits*), l'intersection est calculée entre les traits sémantiques du contexte et les traits sémantiques des définitions candidates. Lorsque l'algorithme est appliqué aux contextes lexicaux (méthode appelée *LeskLex*), l'intersection est calculée entre les unités lexicales du contexte et les traits sémantiques des définitions candidates. L'occurrence à désambiguïser est jugée terminologique si l'intersection avec la définition terminologique contient plus d'éléments que l'intersection avec l'ensemble des définitions non terminologiques. L'occurrence est jugée non terminologique dans le cas contraire.

3 Expériences réalisées

3.1 Méthodologie de l'évaluation

Pour mesurer les performances des différentes méthodes de désambiguïsation, et l'impact de l'annotation sémantique, nous utilisons l'indice courant du taux d'exactitude (*accuracy*). Le taux d'exactitude provient d'une adaptation des métriques de *précision/rappel/F-Mesure* au champ de la désambiguïsation lexicale (Navigli 2009). Les méthodes que nous évaluons se caractérisent par une *couverture* (Nombre de réponses produites / nombre de réponses attendues) de 100 % et, par conséquent, les mesures *précision/rappel/F-Mesure* sont égales et correspondent alors au *taux d'exactitude* (Schwab et *al.* 2013). Ce taux est calculé pour chaque candidat et pour chaque méthode de désambiguïsation testée (avec ou sans annotation sémantique ; textométrie vs. Lesk).

[11] Le recours au calcul de spécificité sur les traits sémantiques pourrait de plus nous permettre de qualifier les emplois terminologiques et non terminologiques de chaque candidat terme. Ceci n'a cependant pas encore été réalisé car cela suppose une analyse plus approfondie de traits sémantiques : différencier le métalangage lexicographique (par exemple « action de *verbe* », « celui qui ») ; affiner le traitement de traits poly-lexicaux (par exemple « faire référence à ») ; classer les traits sémantiques restants en traits inhérents vs. afférents, classifiants vs. spécifiques, etc.

[12] Les définitions à contenu terminologique qui sont présentes dans le TLFi ont été supprimées afin que les définitions du dictionnaire ne relèvent que de la langue générale.

taux d'exactitude = Nombre de réponses correctes / Nombre de d'occurrences du candidat terme

3.2 Jeu de test

Nous l'avons mentionné dans la section (2.1) consacrée aux données de travail et au corpus de référence, le ratio des occurrences validées sur le nombre total d'occurrences d'un candidat terme permet de classer l'ensemble des candidats annotés selon leur taux d'ambiguïté et leur tendance terminologique. Dans le corpus de référence, certaines combinaisons ne sont pas représentées, ce qui est logique : par exemple, on ne trouvera aucun candidat qui soit à la fois très ambigu et très terminologique. Les candidats termes du jeu de test ont été choisis parmi les combinaisons existantes et en fonction des cas d'ambiguïté qu'ils représentent (cf. exemples issus du corpus, section 1). Le tableau (3) ci-dessous résume les informations quantitatives utilisées pour la sélection des candidats termes du jeu de test. L'ensemble des taux d'exactitude obtenus sur le jeu de test est résumé dans le tableau (4) dans lequel les candidats termes sont ordonnés par taux d'ambiguïté croissant.

DONNÉES DE RÉFÉRENCE			
Jeu test	Ambiguïté	Termino	Fréquence
adjectif	1,88	98,12	212
hypothèse	4,22	4,22	166
patient	5,38	5,38	93
locuteur	13,04	89,96	168
exemple	15,34	15,34	567
objet	22,51	22,51	191
corpus	24,00	76,00	550
expression	28,16	28,16	226
argument	30,00	70,00	90
référence	30,28	30,28	218
énoncé	39,31	60,69	173
définition	45,36	54,64	183
Moyenne	**21,62**	**46,28**	**236**
Écart typeP			

Tableau 3 : Candidats termes du jeu de test décrits par leur taux d'ambiguïté , leur tendance terminologique et leur fréquence

RÉSULTATS DES 4 MÉTHODES					
SpecLex	SpecTraits	LeskLex	LeskTraits	Moyenne	Écart typeP
97,44	**97,44**	30,77	94,36	80,00	*28,45*
92,99	**96,18**	82,80	76,43	87,10	*7,90*
96,70	95,60	64,84	94,51	87,91	*13,35*
93,29	**94,51**	58,54	92,68	84,76	*15,15*
68,11	**84,53**	59,25	26,04	59,48	*21,33*
65,41	62,70	**74,05**	34,59	59,19	*14,81*
91,77	91,57	57,43	66,87	76,91	*15,13*
69,52	54,29	53,81	60,95	59,64	*6,37*
82,56	69,77	32,56	41,86	56,69	*20,26*
48,69	53,93	60,21	**63,87**	56,68	*5,82*
73,33	**87,27**	53,33	58,79	68,18	*13,23*
47,73	32,39	62,50	**68,18**	52,70	*13,90*
77,30	76,68	57,51	64,93	69,10	
17,12	*20,63*	*14,06*	*21,75*	12,78	

Tableau 4 : Taux d'exactitude obtenus par méthode pour les candidats termes du jeu de test (en gras, les taux d'exactitude les plus élevés)

L'objectif de notre expérience est d'évaluer dans quelle mesure l'annotation sémantique décrite dans cet article améliore ou non le taux d'exactitude de la procédure automatique de désambiguïsation terminologique, c'est donc dans ce sens que les premiers résultats détaillés sont présentés dans la section (3.2.2). Afin de situer ces premiers résultats, nous les comparons avec une reproduction de l'algorithme de Lesk appliqué aux contextes lexicaux et aux contextes enrichis sémantiquement (section 3.2.3).

3.2.1 Mesure de l'apport de l'annotation en traits sémantiques TLFi & WiktionnaireX V2 sur le jeu de test

La première tendance générale qui se dégage était prévisible : le taux d'exactitude décroît lorsque le taux d'ambiguïté augmente (figure 4) : les moins bonnes performances apparaissent avec les candidats les plus ambigus *exemple*, *objet*, *argument*, *définition*, *référence* et *expression*. La seconde l'était aussi : le taux d'exactitude est peu sensible à la tendance terminologique des candidats termes (figure 3).

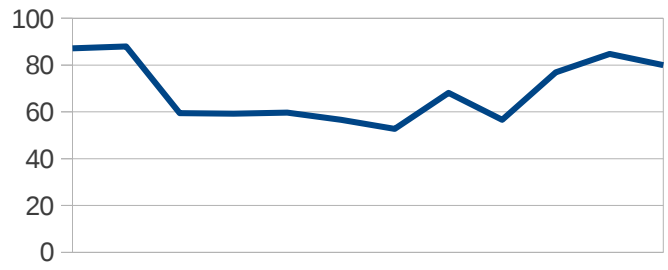


Figure 3: Moyenne des taux d'exactitude par candidats classés par tendance terminologique croissante

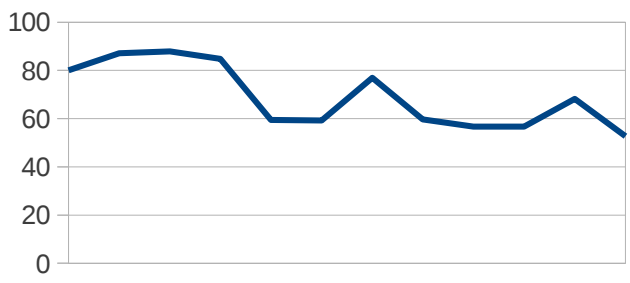


Figure 4: Moyenne des taux d'exactitude par candidats classés par taux d'ambiguïté croissant

Dans la mesure où le taux d'ambiguïté est le paramètre discriminant, c'est en fonction de celui-ci que nous présenterons la suite des résultats de l'expérience. Les courbes suivantes (figure 5) montrent l'évolution du taux d'exactitude en fonction de la mise en œuvre de la désambiguïsation sur des contextes lexicaux ou enrichis sémantiquement. Le premier constat que l'on peut faire est que l'annotation sémantique ne semble pas apporter d'amélioration sensible sur la moyenne des performances de la tâche de désambiguïsation terminologique. Cependant, dans certains cas, par exemple, *exemple* ou *énoncé* dans le jeu de test, on observe une performance supérieure et une performance très légèrement supérieure avec *référence*. Une analyse détaillée de l'ensemble des décisions divergentes sur ces trois candidats permettra dans la suite de nos travaux de mieux comprendre qualitativement l'influence de l'annotation sémantique.

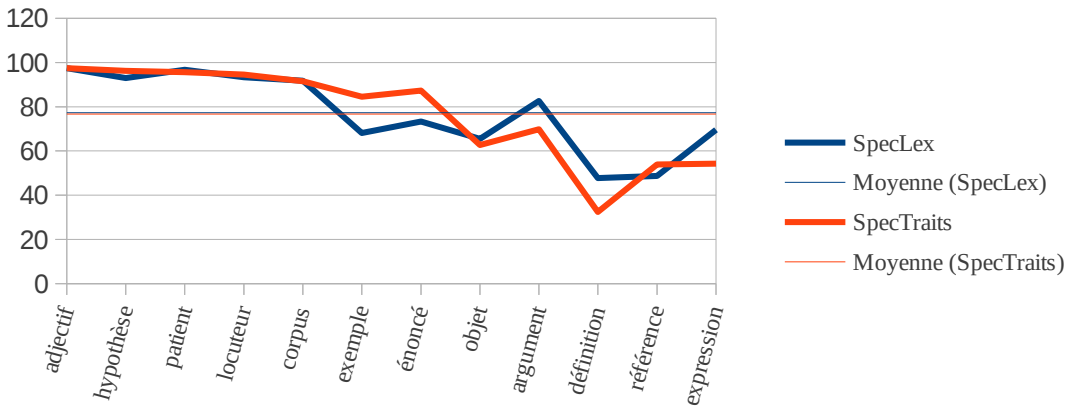


Figure 5: Taux d'exactitude par candidat pour la méthode textométrique et selon le type de contextes (lexicaux, SpecLex, ou annotés en traits sémantiques, SpecTraits)

Une analyse quantitative entre les méthodes SpecLex et SpecTraits est faite dans la section 3.3. Elle est appliquée par la suite parce qu'elle est appliquée non plus seulement au jeu de test mais à l'ensemble de candidats termes et de leurs occurrences.

3.2.2 Comparaison avec l'algorithme de Lesk sur le jeu de test

Pour situer les résultats obtenus par la méthode textométrique, nous avons implémenté la méthode de Lesk dans sa version initiale pour déterminer les taux d'exactitude obtenus à l'aide de cet algorithme fondateur dans le contexte particulier des travaux que nous développons. L'algorithme a été testé avec les contextes lexicaux (figure 6) et avec les contextes annotés en traits sémantiques (figure 7).

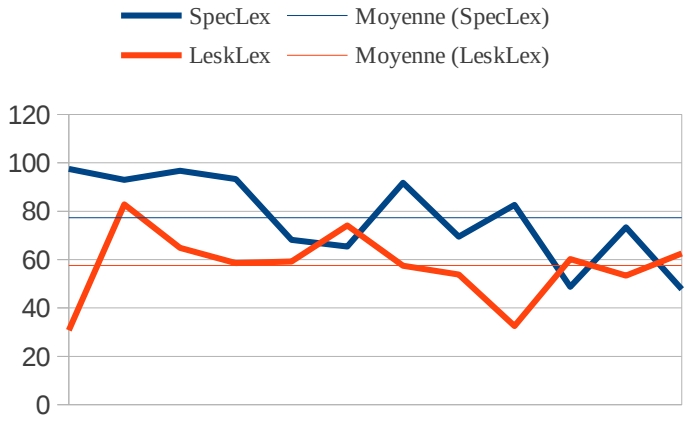


Figure 6: Comparaison des taux d'exactitude par candidat pour la méthode textométrique et pour l'algorithme de Lesk sur les contextes lexicaux

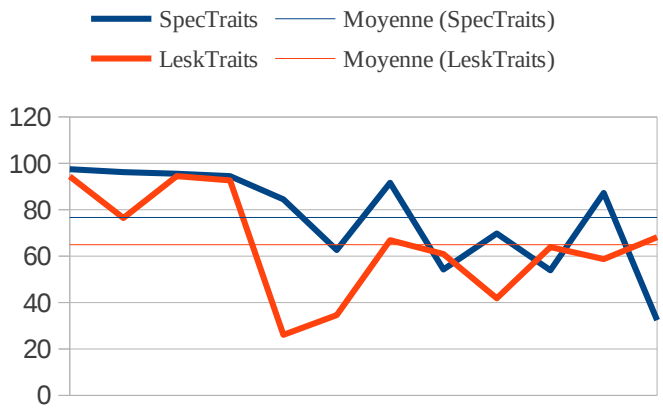


Figure 7: Comparaison des taux d'exactitude par candidat pour la méthode textométrique et pour l'algorithme de Lesk sur les contextes annotés en traits sémantiques

L'amélioration apportée par l'approche textométrique est visible : les courbes SpecLex et SpecTraits se situent au-dessus des courbes LeskLex et LeskTraits respectivement. Cependant, les faibles résultats de Lesk demandent à être explicités. Par rapport à l'état de l'art en effet (Schwab et *al.* 2013) ou (Navigli et *al.* 2007), les performances de l'algorithme de Lesk sont nettement moindres : taux moyen d'exactitude de 77 % avec une implémentation simple de l'algorithme de Lesk dans (Schwab et *al.* 2013 : 105) par exemple. La meilleure moyenne du taux d'exactitude que nous obtenons avec Lesk est de 64,93 % avec les contextes annotés en traits sémantiques.

Une explication possible repose sur la nature de l'ambiguïté que nous cherchons à résoudre. Bien que nous trouvant théoriquement dans une position beaucoup plus favorable comparativement à la difficulté que représente la désambiguïsation lexicale sémantique dans son entier, nous faisons face à des cas d'ambiguïté très fins. En témoignent les deux exemples ci-dessous du candidat *définition* : le premier montre un emploi jugé terminologique par l'annotation manuelle, le second emploi relève d'un emploi non terminologique.

> [Terminologique] : *Aujourd'hui, des relations entre termes sont de plus en plus souvent intégrées aux terminologies pour compléter les* ***définitions*** *en langage naturel.* Les relations sémantiques : du linguistique au formel - Aussenac-Gilles N. et Séguéla P. (2000). Cahiers de grammaire (25)
>
> [Non terminologique] : *La* ***définition*** *des termes est désormais le résultat d'une analyse systématique de l'usage des termes en corpus.* Les relations sémantiques : du linguistique au formel - Aussenac-Gilles N. et Séguéla P. (2000). Cahiers de grammaire (25)

Ce type d'ambiguïté terminologique entre, par ailleurs, en résonance avec une ambiguïté courante en sémantique lexicale : le lien métonymique entre une interprétation résultative et une interprétation processive que l'on rencontre souvent avec les noms déverbaux, ce qui est le cas du candidat *définition*. Le même cas de figure se présente le candidat *expression*.

> [Terminologique] : *[…] les* ***expressions*** *du type le jour suivant […]* Validation d'une méthodologie pour l'étude des marqueurs de la segmentation dans un grand corpus de texte - Piérard S. et Begsten Y. (2007). TAL(47/2)
>
> [Non terminologique] : *L'****expression*** *de telle ou telle relation […]* Variabilité des outils de TAL et genre textuel : cas des patrons lexico-syntaxiques – Jacques M.-P. Et Assenac-Gilles N. (2006). TAL (47)

Pour ce type d'ambiguïté, la tâche de la compétition SemEval 2007 (Navigli et *al.* 2007) la plus proche est la tâche 8 : *Metonymy Resolution*. La tâche 8 de SemEval 2007 avait pour but de différencier trois niveaux de désambiguïsation (de la plus grossière à la plus fine) entre les sens métonymiques apparaissant avec les noms de lieu et les noms d'organisation. Pour chacune des catégories (lieux ou organisation), (Markert et Nissim 2007) ont défini la baseline du taux d'exactitude à 79,4 % pour la distinction entre les sens des noms de lieu et à 61,4 % pour la distinction entre les sens des noms d'organisation. Par rapport à ces taux d'exactitude, les deux variantes de la méthode Lesk telle que nous

l'avons implémentée (taux d'exactitude moyen de 57,51 % pour LeskLex et de 64,93 % pour LeskTraits) se situent autour du taux obtenu pour la distinction entre les sens des noms de lieux.

3.3 Comparaison SpecLex-SpecTraits sur un jeu de données plus étendu

La comparaison entre les résultats obtenus par la désambiguïsation textométrique (SpecLex et SpecTraits) sur le jeu de test nous a amené a conclure que l'annotation sémantique n'apportait pas d'amélioration sensible. Afin de vérifier ce constat sur un jeu de données plus important, nous avons pris deux textes du corpus Scientext (un article de revue et une communication) comme corpus d'évaluation et nous avons passé les méthodes de désambiguïsation par toutes les occurrences de candidats termes extraits par TTC-Termsuite. Dans cette nouvelle expérience, l'ambiguïté globale est plus modeste (10,34/50) et l'échantillon est plus hétérogène au niveau des fréquences. Au total, 2284 occurrences de candidats termes ont été évaluées par rapport à l'annotation manuelle.

	Types	Occurrences	Ambiguïté	SpecLex	SpecTraits
Candidats monolexicaux	339	1903	11,64	79,72	82,71
Candidats polylexicaux	92	381	3,84	80,58	80,05
Total	431	2284	10,34	**79,86**	**82,27**

Tableau 5 : Analyse des données du corpus d'évaluation

Comme le montre le tableau ci-dessus, l'ambiguïté des candidats monolexicaux (11,64/50) est bien supérieure à celle des candidats termes polylexicaux (3,84/50), c'est pourquoi les données du jeu de test (section 3.2) correspondent à des candidats monolexicaux uniquement. Sur le jeu de données plus étendu qui a été utilisé pour cette seconde expérience, on constate une amélioration des taux d'exactitude, que ce soit pour la méthode SpecLex ou la méthode SpecTraits. Cependant, contrairement à ce qu'on avait pu observer sur le jeu de test de la section précédente, on peut noter que l'utilisation des contextes enrichis en traits sémantiques (taux d'exactitude : 82,27% contre 79,86%) apporte cette fois une amélioration sensible par rapport aux résultats observés sur le jeu de test. Un étude approfondie de la complémentarité de ces deux méthodes doit donc être menée pour créer une méthode globale de désambiguïsation terminologique.

4 Conclusion et perspectives

Cet article se situe dans le champ de la désambiguïsation terminologique qui peut se voir *a priori* comme un cas de figure simplifié de désambiguïsation lexicale. La méthode que nous avons évaluée sur un jeu de candidats termes à désambiguïser s'appuie sur l'exploitation des contextes d'occurrences présentés sous deux formes : une version lexicale (contextes lexicaux) et une version enrichie en traits sémantiques (contextes sémantiques). Les résultats obtenus montrent que l'annotation sémantique n'apporte pas d'amélioration sensible pour la moyenne des performances mesurées sur le jeu de test. Cependant, nous avons vu que cette tendance s'inverse lorsque l'expérience est appliquée aux 67 candidats les plus fréquents dans le corpus de travail. Ce constat nous conduit à envisager deux perspectives immédiates : reproduire les expériences sur l'ensemble des candidats termes et analyser en détail les occurrences où les méthodes divergent. En effet, la figure (9) montre l'amplitude des variations possibles avec la méthode textométrique comme avec la méthode inspirée de Lesk. Pour pouvoir envisager une automatisation complète de la tâche de désambiguïsation, l'analyse des divergences permettrait de déterminer comment articuler les méthodes entre elles et dans quels cas privilégier l'une par rapport à l'autre.

Sur le plan des améliorations à apporter à la désambiguïsation elle-même, il serait intéressant de la faire collaborer avec des approches distributionnelles car celles-ci permettraient très probablement d'écarter d'emblée un certain nombre de cas massifs où l'occurrence d'un candidat terme fait partie d'une locution comme *par définition, faire référence à*, etc. Sur le plan des méthodes statistiques utilisées, une comparaison avec des méthodes de type SVM (*Support Vector Machines*) serait pertinente. Ces méthodes réduisent en effet la tâche de désambiguïsation à un problème de classification, en obtenant de très bons résultats par rapport à d'autres méthodes (Lee and Ng 2002 ; Navigli 2009).

Sur le plan de l'annotation sémantique enfin, nous envisageons de poursuivre nos travaux dans deux directions complémentaires. La première voie s'intéressera à deux manières possibles de limiter la dispersion de l'information sémantique due à l'utilisation des définitions d'une ressource lexicale sémantique : (1) nous appuyer sur l'hypothèse que le premier mot de même catégorie que l'unité lexicale à annoter joue le rôle d'un hyperonyme ou plutôt réfère au genre prochain dans la terminologie de la TST ; (2) utiliser les domaines d'usage qui figurent dans les définitions du TLFi. La seconde évolution possible pour l'annotation sémantique visera à utiliser une autre ressource lexicale, à savoir le

WordNet libre pour le français, la ressource Wolf telle qu'elle a été mise à disposition de la communauté par Benoît Sagot[13]. À l'aide de Wolf, nous envisageons une annotation en synsets[14] qui pourrait être réalisée de deux manières : (1) une annotation comparable à celle que nous avons développée dans cet article mais avec les synsets ou concepts possibles selon une correspondance des formes lemmatisées entre texte et ressource lexicale ; (2) une annotation désambiguïsante où le concept représentant un mot d'un article est identifié selon son contexte dans l'article. Des travaux d'indexation de sites web s'appuyant sur l'exploitation de WordNet sont déjà proposés (Desmontils et Jacquin 2001, Benyahia et *al*. 2009).

Remerciements

Nos vifs remerciements aux relecteurs/relectrices pour leurs remarques et suggestions qui ont permis d'améliorer cette proposition ; toute imprécision, erreur restante est bien sûr de notre entière responsabilité. Nos chaleureux remerciements à l'équipe du projet TermITH, au sein du laboratoire (en particulier Bertrand Gaiffe, Benjamin Husson, Etienne Petitjean, Jean-Marc Humbert et Sandrine Ollinger) ; au sein des partenaires du projet, en particulier Béatrice Daille (LINA), Agnès Tutin, Marie-Paule Jacques et Sylvain Hatier (LIDILEM), Claire François, Sabine Barreaux et Flora Badin (INIST), Yannick Toussaint et Felipe Melo-Mora (LORIA), Laurent Romary et Patrice Lopez (INRIA Saclay). Nos remerciements enfin à l'ANR pour le soutien financier accordé au projet TermITH (ANR-12-CORD-0029) ainsi qu'à nos tutelles (CNRS, INRIA, Université de Grenoble3, de Lorraine, de Nantes).

Références

Aubin S. and Hamon T. (2006) Improving Term Extraction with Terminological Resources. In *Advances in Natural Language Processing (5th International Conference on NLP, FinTAL)*.

Aussenac-Gilles N., Condamines A. (2009). Marqueurs de relations, genre textuel, structures syntaxiques. In Minel, J.-L. (Ed.), *Filtrage sémantique*, 115-149. Paris: Hermes/Lavoisier.

Baneyx A., Malaisé V., Charlet J., Zweigenbaum P. et Bachimont B. (2005). Synergie entre analyse distributionnelle et patrons lexico-syntaxiques pour la construction d'ontologies différentielles. Dans Actes de la conférence *TIA-2005*, 12 pages. http://estime.spim.jussieu.fr/~jc/Files/BaneyxTIA2005.pdf [dernier accès (*d.a.*) 11/02/14]

Benyahia K., Lehireche A. et Latreche A. (2009). Annotation sémantique de pages Web. In Actes de la seconde *Conférence Internationale sur l'Informatique et ses Applications*, Saida, Algérie, 3-4 Mai, 8 pages. http://ceur-ws.org/Vol-547/54.pdf [pages consultées le 11/02/14]

Bourigault D., Jacquemin C. et l'HommeM.C. (2001). Recent Advances in Computationnal Terminology. John Benjamins :Amsterdam.

Bourigault D., Slozdian M. (1999). Pour une terminologie textuelle. Revue *Terminologies Nouvelles* 19 (Actes de la conférence *TIA 1999*), 29-32. http://www.rifal.org/cahiers/rint19/rint19.pdf. [(*d.a.*) 11/02/14]

Camacho collados J., Billami M. B., Jacquey E., Kister L. (2014). Approche statistique pour le filtrage terminologique des occurrences de candidats termes en texte intégral. Dans Actes des *Journées internationales d'Analyse statistique des Données Textuelles (JADT)*, Paris, France, 3-6 Juin

Claveau V. et L'Homme M.C. (2005). Apprentissage par analogie pour la structuration de terminologies - utilisation comparée de ressources endogènes et exogènes. Dans Actes de la conférence *TIA-2005*, 12 pages. http://www.irisa.fr/texmex/people/claveau/publis/Claveau-LHomme-tia05.pdf. [(*d.a.*) 11/02/14]

Condamines A., Péry-Woodley M.P. (2007). Linguistic markers of semantic and textual relations. In Alamargot, D., Terrier, P. & Cellier, J.-M. (Eds.), *Written documents in the workplace. Studies in Writing*. 3-16. Amsterdam: Elsevier.

Daille B. (1994). Approche mixte pour l'extraction de terminologie : statistique lexicale et filtres linguistiques, Thèse en informatique fondamentale, Université Paris 7.

Daille B. (2003). Conceptual structuring through term variations. In *Proceedings of the ACL2003 Workshop on Multiword Expressions : Analysis, Acquisition and Treatment*, Bond E., Kohonen A. Carthy D.M. et Villalencio A. (eds), 9-16.

Daille B., Kageura K., Nakagawa H., Chien L.-F. (eds). (2004). Recents Trends in Computational Terminology, *Terminology*, 10(1). ISSN 0929-997.

[13] WOLF : WordNet Libre du Français, http://alpage.inria.fr/~sagot/wolf.html [page consultée le 04/02/2014].

[14] Un synset (ou un concept) correspond à un ensemble de mots qu'on peut les qualifier de synonymes entre eux représentant un sens très précis à l'aide d'une définition.

DAILLE B., JACQUIN CH., MONCEAUX L., MORIN E. et ROCHETEAU J. (2011). TTC TermSuite : une chaîne de traitement pour la fouille terminologique multilingue. Démonstration au cours de la conférence *TALN 2011*. http://www.lirmm.fr/taln2011/DEMOS/DEMO_Daille_UnivNantes.pdf. [(*d.a.*) 11/02/14]

DESMONTILS E ; ET JACQUIN C. (2001). Des ontologies pour indexer un site Web. In *Journées francophones d'ingénierie des connaissances (IC)*, Jean Charlet (ed;), Presses Universitaires de Grenoble (PUG), Grenoble, 25-28 juin, 131-146,

DROUIN P. (2003). Term extraction using technical corpora as a point of leverage. In *Terminology* 9(1), 99-117.

DUBOIS J., GIACOMO M., GUESPIN L., MARCELLESI C., MARCELLESI J-B. ET MÉVEL J-P. (2012) DICTIONNAIRE DE LINGUISTIQUE. LAROUSSE.

DUCROT O. ET SCHAEFFER J-M. (1999) NOUVEAU DICTIONNAIRE ENCYCLOPÉDIQUE DES SCIENCES DU LANGAGE. ESSAIS, 832 PAGES.

GRABAR N., ZWEIGENBAUM P. (2004). Lexically-based terminology structuring. In *Terminology*, 10(1), 23-54. Résumé : http://www.ingentaconnect.com/content/jbp/term/2004/00000010/00000001/art00002. [(*d.a.*) 12/02/14]

HABERT B. (2005). Portrait de linguiste(s) à l'instrument », Texto! vol. X, n° 4, 2005.

HEIDEN S., MAGUÉ J-P., PINCEMIN B. (2010). TXM : Une plateforme logicielle open-source pour la textométrie – conception et développement. In Actes de la conférence *JADT 2010 : 10th International Conference on the Statistical Analysis of Textual Data*, Rome : Italie : 12 pages http://halshs.archives-ouvertes.fr/docs/00/54/97/79/PDF/Heiden_al_jadt2010.pdf. [pages consultées le 11/02/14]

JACQUES M-P., AUSSENAC-GILLES N. (2006). Variabilité des performances des outils de TAL et genre textuel. Cas des patrons lexico-syntaxiques. Dans : *Traitement Automatique des Langues*, 47(1), 11-32. http://www.atala.org/Variabilite-des-performances-des [(*d.a.*) 11/02/14]

LAFON P. (1980). Sur la variabilité de la fréquence des formes dans un corpus. *Mots* (1), 127-165.

LEE, K.Y. AND NG, H. T. (2002) An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation, In Actes de la conférence *On Empirical Methods in Natural Language Processing* (EMNLP), Philadelphia, USA, 41–48.

LESK M. (1986). Automatic sense disambiguation using MRD : how to tell a pine cone fom an ice cream cone. In *Proceeding of SIGDOC' 86, ACM*, New York, USA, 24-26.

L'HOMME M.C. (2004a). Adjectifs dérivés sémantiques dans la structuration de terminologies. Dans Actes de la conférende *Terminologie, ontologie et représentation des connaissances*, Université Jean-Moulin Lyon-3, 22-23 janvier 2004. 6 pages. http://olst.ling.umontreal.ca/pdf/lhomme-lyon2003.pdf. [(*d.a.*) 11/02/14]

L'HOMME M.C. (2004b). *La terminologie : principes et techniques*, Montréal : Presses de l'Université de Montréal.

MANSER M. (2012). État de l'art sur l'acquisition de relations sémantiques entre termes : contextualisation des relations de synonymie. Dans Actes de la conférence *JEP-RECITAL 2012*, 163-175. http://aclweb.org/anthology//F/F12/F12-3013.pdf. [(*d.a.*) 11/02/14]

MARKERT K., NISSIM M. (2007). SemEval-2007 Task 08 : Metonymu Resolution at SemEval-2007. In *Proceedings of the International Workshop on Semantic Evaluations (SemEval-2007)*, 36-41, Prague, Association for Computational Linguistics. http://acl.ldc.upenn.edu/W/W07/W07-2007.pdf [(*d.a.*) 12/02/2014]

NAMER F., ZWEIGENBAUM P. (2004). Acquiring meaning for French medical terminology: contribution of morphosemantics. In Marius Fieschi, Enrico Coiera, and Yu-Chuan Jack Li, editors, *Proceedings 10th World Congress on Medical Informatics*, volume 107 of *Studies in Health Technology and Informatics*, 535-539, Amsterdam, 2004. IOS Press. http://www.ncbi.nlm.nih.gov/pubmed/15360870?dopt=Abstract [(*d.a.*) 11/02/14]

NAVIGLI, R. (2009).Word Sens Disambiguation : A Survey, *ACM Computing Surveys*, 41(2).

NAVIGLI R., LITKOWSKI K.B. ET HARGRAVES O. (2007). SemEval-2007 Task 07 : Coarse-Greined Anglish All-words Task. In *Proceedings of the International Workshop on Semantic Evaluations (SemEval-2007)*, 30-35, Prague, Association for Computational Linguistics.

NAVIGLI R. ET LAPATA, M. (2010) An Experimental Study of Graph Connectivity for Unsupervised Word Sens Disambiguation, In *IEEE Trans. Patter Anal. Mach. Intell.* , vol.32, pp. 678-692.

NEVEU F. (2004) DICTIONNAIRE DES SCIENCES DU LANGAGE. ARMAND COLIN, 316 PAGES.

PÉRINET A. , HAMON H. (2013). Hybrid acquisition of semantic relations based on context normalization in distributional analysis. Dans Actes de la conférence *TIA-2013*, 113-120. https://lipn.univ-paris13.fr/tia2013/Proceedings/actesTIA2013.pdf. [(*d.a.*) 11/02/14]

Poibeau T. (2005). Parcours interprétatifs et terminologie. Dans *Actes TIA 2005*. Rouen.

Rastier F., Valette M. (2009). De la polysémie à la néosémie. Dans *Texto!* XIV(1), 1-18. http://www.revue-texto.net/docannexe/file/2119/last_rastier_valette_polysemie.pdf

Retenauer C. 2012. Vers un traitement automatique de la néosémie : approche textuelle et statistique. Université de Lorraine., Thèse de doctorat en sciences du langage.

Sajous F., Navarro E., Gaume, B., Prévot, L., Chudy, Y. (2013). Semi-Automatic Enrichment of Crowdsourced Synonymy Networks: The WISIGOTH system applied to Wiktionary. *Language Resources & Evaluation*, 47(1), 63-96.

Schwab D., Goulian J., Tchechmedjiev A. (2013). Désambiguïsation lexicale de textes : efficacité qualitative et temporelle d'un algorithme à colonies de fourmis, In *TAL* 54(1), 99-138.

Toussaint Y., Namer F., Daille B., Jacquemin C., Royauté J., Hathout N. (1998). Une approche linguistique et statistique pour l'analyse de l'information en corpus. Dans *Actes de la conférence TALN'98*, ATALA, Paris, France.

Valette M., Estacio-Moreno A., Petitjean E., Jacquey E. (2006). Éléments pour la génération de classes sémantiques à partir de définitions lexicographiques. Pour une approche sémique du sens. Dans *Actes de la 13ème conférence sur le traitement automatique des langues naturelles (TALN 06)*, P. Mertens, C. Fairon, A. Dister, P. Watrin (éds). *Cahiers du CENTAL*, 2.1, UCL Presses Universitaires de Louvain (1), 357-366.

Identification des noms sous-spécifiés, signaux de l'organisation discursive

Charlotte Roze[1, 2] Thierry Charnois[3] Dominique Legallois[2] Stéphane Ferrari[1]
Mathilde Salles[2]
(1) GREYC, Université de Caen Basse-Normandie, Campus 2, 14000 Caen, France
(2) CRISCO, Université de Caen Basse-Normandie, Campus 1, 14000 Caen, France
(3) LIPN, Université Paris 13 Sorbonne Paris Cité, 93430 Villetaneuse, France
{charlotte.roze,dominique.legallois, stephane.ferrari, mathilde.salles}@unicaen.fr,
thierry.charnois@lipn.univ-paris13.fr

Résumé. Dans cet article, nous nous intéressons aux *noms sous-spécifiés*, qui forment une classe d'indices de l'organisation discursive. Ces indices ont été peu étudiés dans le cadre de l'analyse du discours et en traitement automatique des langues. L'objectif est d'effectuer une étude linguistique de leur participation à la structuration discursive, notamment lorsqu'ils interviennent dans des séquences organisationnelles fréquentes (e.g. le patron *Problème-Solution*). Dans cet article, nous présentons les différentes étapes mises en oeuvre pour identifier automatiquement ces noms en corpus. En premier lieu, nous détaillons la construction d'un lexique de noms sous-spécifiés pour le français à partir d'un corpus constitué de 7 années du journal *Le Monde*. Puis nous montrons comment utiliser des techniques fondées sur la fouille de données séquentielles pour acquérir de nouvelles constructions syntaxiques caractéristiques des emplois de noms sous-spécifiés. Enfin, nous présentons une méthode d'identification automatique des occurrences de noms sous-spécifiés et son évaluation.

Abstract. In this paper, we focus on *shell nouns*, a class of items involved in the signaling of discourse organisation. These signals have been little studied in Natural Language Processing and within discourse analysis theories. The main goal is to study their participation to discourse organisation, especially when they occur in Problem-Solution patterns. In this paper, we present the different steps involved in shell nouns identification of these nouns. First, we present the lexical acquisition of shell nouns from a large corpus. Second, we show how a method based on the extraction of sequential patterns (sequential data mining techniques) allows to discover new syntactic patterns specific to the use of shell nouns. Finally, we present a shell nouns identification system that we evaluate.

Mots-clés : noms sous-spécifiés, motifs séquentiels, structure discursive.

Keywords: shell nouns, sequential patterns, discourse structure.

1 Introduction

Le travail que nous présentons s'inscrit dans le cadre de l'analyse de l'organisation discursive et de l'étude des indices linguistiques de cette organisation. Parmi ces indices, une classe d'items a reçu jusqu'ici peu d'attention en Traitement Automatique des Langues et dans les théories d'analyse du discours telles que la RST (Mann et Thompson, 1988) ou la SDRT (Asher et Lascarides, 2003) : les *noms sous-spécifiés* (Legallois, 2008), appelés *"shell nouns"* (Schmid, 2000) ou *"signalling nouns"* (Flowerdew, 2003) dans les travaux sur l'anglais.

Les noms sous-spécifiés (désormais NSS) sont des noms comme *problème*, *idée* ou *objectif*, ayant non seulement la capacité de référer à des entités abstraites [1] décrites par une proposition syntaxique, une phrase, ou des unités discursives plus larges, mais aussi de leur attribuer un label, de caractériser leur contenu, ou leur fonction dans l'organisation du discours dans lequel ils apparaissent. Selon Schmid (2000), ils fonctionnent comme des « coquilles conceptuelles » : ils présentent une certaine incomplétude sémantique, qui est comblée par le contenu des entités auxquelles ils réfèrent. Cette incomplétude leur confère un statut proche de celui de prédicat. L'exemple (1) [2] présente une portion de discours

1. La notion d'entité abstraite a été introduite par (Asher, 1993).
2. Cet exemple est tiré du corpus *Le Monde*.

contenant trois occurrences de NSS : les noms *idée*, *objectif* et *résultat* (en gras). Ces noms réfèrent aux entités abstraites correspondant aux portions de textes entre accolades. Ici, *idée* et *objectif* étiquettent les unités auxquelles ils réfèrent comme comportant l'expression d'un but, et *résultat* l'expression d'une conséquence.

1. L'opérateur [SFR] se dit prêt à équiper, en 2003, 2000 sites capables d'accueillir des stations de base UMTS, ceci dans cinq villes françaises : Marseille, Lyon et Nice et les deux villes pilotes [...] Mais pas question d'étendre le futur réseau UMTS à l'ensemble du territoire. Pour SFR, cette technologie [...] sera réservée aux grandes agglomérations. L'**idée** est d'{offrir une continuité de services} grâce au réseau GPRS [...] SFR poursuit donc son investissement dans le GSM et le GPRS. L'**objectif** est d'{atteindre une couverture de 95 % du territoire fin 2005}, contre 84 % fin 2002. **Résultat**, {SFR accroît en 2003 ses investissements dans le réseau}.

Les NSS forment une classe fonctionnelle : les noms comme *problème* ou *objectif* n'ont pas la capacité de référer à des entités abstraites dans toutes leurs occurrences. Comme le souligne Schmid (2000), les noms comme *fact* ou *reason* ne sont pas des "shell nouns" grâce à une propriété qui leur est inhérente, ils deviennent des "shell nouns" dans certains de leurs emplois. De même, Legallois (2008) souligne que la notion de NSS s'applique à un type d'emploi nominal et non à une nature nominale. Plus précisément, il existe des constructions syntaxiques caractéristiques des emplois de NSS. Schmid (2000) décrit pour l'anglais un ensemble de patrons syntaxiques accueillant des "shell nouns", dont le patron *N be that–clause* (comme dans *the fact is that* par exemple). Pour le français, Legallois (2008) identifie les *constructions spécificationnelles* comme caractéristiques des emplois de NSS (voir section 2). Pour le français, Legallois et Gréa (2006) analysent les constructions spécificationnelles comme des dispositifs syntaxiques permettant la spécification du contenu indéterminé de ces noms. Ces constructions ont pour forme *Det N être* (*que–clause* | *de–inf*). Les noms comme *problème*, *solution* ou *objectif* possèdent donc la fonction de NSS dans certains de leurs emplois uniquement. Par exemple, en (2), l'occurrence de *problème* ne correspond pas à un emploi en tant que NSS.

2. Pour de nombreux économistes, le nouveau code du travail ne résoudra pas les **problèmes** de productivité et de compétitivité de l'économie portugaise.

Dans le cadre du Traitement Automatique des Langues, ces noms présentent évidemment un intérêt de recherche en ce qui concerne la résolution d'anaphores, et font récemment l'objet des travaux de Kolhatkar *et al.* (2013a,b), qui ont pour objectif principal d'améliorer, pour l'anglais, la résolution de leurs antécédents dans leurs emplois anaphoriques. En effet, ces noms peuvent référer à des unités abstraites de façon cataphorique, comme dans les emplois de *idée* et *objectif* dans l'exemple (1), mais aussi de façon anaphorique, comme l'emploi avec un déterminant démonstratif *ces résultats* en (3). Kolhatkar *et al.* s'intéressent à la tâche d'identification d'antécédents des emplois anaphoriques de "shell nouns", et se heurtent au manque de données annotées pour cette tâche. Kolhatkar *et al.* (2013a) se concentrent sur l'annotation manuelle des occurrences anaphoriques de ces noms par "crowdsourcing", afin de disposer de corpus d'apprentissage pour la tâche de résolution, et Kolhatkar *et al.* (2013b) utilisent comme données d'apprentissage les entités auxquelles réfèrent les "shell nouns" dans leurs emplois cataphoriques, la tâche d'identification étant bien plus aisée dans ce second cas, étant donné qu'elle peut au moins partiellement s'appuyer sur la syntaxe.

3. {Les IRM réalisées montrent une réactivation des zones du cortex moteur généralement dévolues à la main et au coude, sauf pour le troisième patient, dont l'accident était plus ancien.} [...] Comment expliquer ces **résultats** ? « Notre **hypothèse** est que {le fait de voir la main en mouvement réintroduit une cohérence dans le cerveau avec la représentation que le patient a de son corps} », avance Angela Sirigu.

Dans le présent travail, nous laissons de côté la question de l'identification des entités auxquelles les NSS réfèrent, et souhaitons nous intéresser à leur rôle dans l'organisation discursive, jusqu'ici peu étudié dans sa globalité — certains travaux, comme ceux de Vergez-Couret *et al.* (2011) concernant *pour deux raisons*, étudient le rôle discursif de cas particuliers d'emplois de NSS. Les NSS semblent pouvoir participer à l'organisation discursive, soit en spécifiant la fonction ou le contenu d'une unité au sein d'une unité discursive plus large (texte, paragraphe), soit en signalant un lien entre deux unités. Ils peuvent également constituer des marqueurs de frontières de segments textuels, ou des marqueurs de changement thématique (Schmid, 2000). Malgré ces propriétés, les NSS ont jusqu'ici suscité peu d'intérêt dans les recherches en analyse (linguistique ou automatique) du discours. L'objectif du travail présenté ici est d'effectuer une identification en corpus satisfaisante des occurrences de NSS en français, en vue d'une étude linguistique de leur participation à la structuration discursive, notamment lorsqu'ils interviennent dans des *séquences organisationnelles* fréquentes, c'est-à-dire des séquences du type *Problème–Solution* ("*Problem-Solution patterns*"), largement étudiées dans la littérature (Flowerdew, 2008). L'idée est d'identifier ces séquences organisationnelles fréquentes à l'aide des signaux que constituent les NSS (ou encore les connecteurs), et d'examiner les liens qu'entretiennent la présence de ces séquences organisationnelles avec la structure discursive, et plus particulièrement les relations de discours.

Cet article est organisé comme suit : à la section 2, nous présentons la construction d'un lexique de ces noms pour le français ; à la section 3, nous présentons une méthode d'identification des patrons syntaxiques caractéristiques des emplois

de NSS, qui s'appuie sur des techniques de fouille de données ; à la section 4, nous présentons une méthode d'identification automatique des NSS en corpus et son évaluation. Pour terminer, nous présentons les conclusions et perspectives de ce travail (section 5).

2 Construction d'un lexique

Dans cette section, nous présentons la construction/sélection d'un lexique de NSS pour le français. Cette construction est effectuée à partir d'un corpus constitué de 7 années d'articles du journal *Le Monde* (voir section 2.1 pour une description du corpus et des pré-traitements). Elle s'appuie sur l'extraction des occurrences de constructions syntaxiques identifiées dans la littérature comme étant caractéristiques des emplois de NSS, les constructions spécificationnelles (voir section 2.2). À la section 2.3, nous présentons les résultats de l'extraction, et la sélection du lexique, qui repose sur la fréquence d'apparition des noms dans les constructions spécificationnelles.

2.1 Corpus de travail et pré-traitements

Nous travaillons sur un corpus constitué de 7 années d'articles du journal *Le Monde* (de 2000 à 2006), ce qui correspond à 352 265 documents, 7 121 931 phrases et 165 097 356 tokens. Le corpus de départ est au format XML. À chaque document sont associées des métadonnées comme un secteur (une, société, france, débat, art, etc.), des catégories (critique, chiffre, mutation, chronique, important, opinion, etc.), un auteur, une date de parution, des mises à jour, etc. Au sein du texte contenu dans le document, les frontières de paragraphes sont identifiées [3].

Pour construire le lexique de NSS, nous nous appuyons sur l'identification de structures syntaxiques spécifiques, à savoir les constructions spécificationnelles (voir section suivante). Le pré-traitement du corpus comprend donc une phase d'analyse syntaxique. Nous utilisons l'analyseur syntaxique en dépendances Bonsaï (Candito *et al.*, 2010). Pour cela, nous convertissons les documents XML du corpus à un format pouvant être traité par l'analyseur en dépendances syntaxiques Bonsaï. Nous conservons un certain nombre d'informations pouvant être utiles pour nos expériences, comme les frontières de paragraphes, les métadonnées concernant le secteur et les catégories attribuées aux documents. Bonsaï prend en entrée du texte brut, opère une tokenisation (identification des composés), puis un étiquetage en parties du discours faisant appel au Melt Tagger (Denis et Sagot, 2009). L'analyse syntaxique proprement dite est effectuée par le MaltParser (Nivre *et al.*, 2006). Dans la phase de tokenisation du corpus, nous avons ajouté l'ensemble des connecteurs discursifs aux formes composées identifiées comme tokens avant l'analyse syntaxique, et également certaines formes composées auxquelles appartiennent des NSS comme *point*, mais ne correspondant pas (de façon autonome au moins) à des emplois comme NSS (*à ce point*, *point de vue*).

2.2 Méthode d'extraction

Comme nous l'avons vu précédemment, (Schmid, 2000) identifie comme caractéristiques des emplois de "shell nouns" en anglais les structures syntaxiques correspondant au patron : *Det N be* (*that–clause* | *wh–clause* | *to–inf*). Pour le français, les structures caractéristiques identifiées par (Legallois, 2008) sont les constructions dites spécificationnelles, qui couvrent des phrases copulatives dans lesquelles l'objet du verbe *être* est une complétive ou un infinitif. Ces constructions sont décrites par le patron suivant :

$$\textit{Det N } (\varnothing \mid \textit{ce}) \textit{ être } (\textit{que–clause} \mid \textit{de–inf}).$$

On trouve des exemples de constructions spécificationnelles dans les phrases en (4), avec une complétive, et (5), avec un infinitif. Parmi les constructions spécificationnelles, on trouve également des pseudo-clivées, comme en (6).

4. Le **risque** est que ce genre de comportement rappelle de bien mauvais souvenirs.
5. La **question** est de savoir ce que l'on prend comme élément de référence.
6. Le **problème**, c'est que les Occidentaux ne comprennent pas leur mentalité.

3. C'est une des raisons pour lesquelles nous avons choisi ce corpus. Nous verrons dans les perpectives que disposer de ce type de d'informations concernant l'organisation du texte pourra nous aider dans l'identification des séquences organisationnelles.

La notion de construction spécificationnelle s'appuie sur la classification de Higgins (1979) des phrases copulatives. Higgins identifie quatre types de phrases copulatives : prédicative (*cette voiture est rapide*), identificatrice (*la dame avec un chapeau, c'est Madame Dupont*), identité (*l'étoile du matin est l'étoile du soir*), spécificationnelle (*ce que je voudrais, c'est que tu gares la voiture*). Ce type de construction est également abordé par Apothéloz (2008), qui s'intéresse aux constructions spécificationnelles — pour lesquelles il emploie le terme de constructions identificatives — et plus particulièrement aux pseudo-clivées. Parmi ces constructions, Apothéloz relève notamment des cas dans lesquels le segment gauche est un adjectif nominalisé (*important*, *mieux*, *pire*), et des cas dans lesquels le segment gauche est un nom. Parmi les noms observés dans ces constructions, il observe des « lexèmes évaluatifs (*difficulté*, *problème*, *ennui*) ou des hyperonymes servant à construire un syntagme évaluatif (*une* ***chose*** *frappante*, *le* ***truc*** *sur lequel je ne suis pas d'accord*) » et des « lexèmes se rapportant à l'activité langagière, notamment dans ses aspects argumentatifs et explicatifs (cf. *remarque*, *hypothèse*, *preuve*, *raison*, *proposition*) ».

Nous reprenons ici l'idée de Legallois (2008), qui est de s'appuyer sur le repérage en corpus des constructions spécificationnelles pour identifier un lexique de NSS, en étendant la taille du corpus d'extraction. Nous rassemblons sous l'étiquette NSS des noms entrant dans d'autres constructions que les constructions spécificationnelles, mais nous appuyons sur celles-ci pour construire un lexique à partir duquel travailler. Nous effectuons un repérage des constructions spécificationnelles sur l'ensemble du corpus décrit dans à la section 2.1. Pour identifier les constructions, nous recherchons les contextes correspondant au schéma de la figure 1 dans les analyses en dépendances syntaxiques des phrases du corpus. Une fois ce contexte repéré dans une phrase du corpus, l'extracteur vérifie que plusieurs contraintes sur les dépendants de *être* sont respectées : il ne peuvent pas être des participes passés ou des adjectifs, il ne peuvent pas être des pronoms réflexifs, et ne peuvent pas non plus être des prépositions (*être en mesure de* analysé avec *en* dépendant de *être*, et *de* autre dépendant de *être*). En revanche, les adverbes sont admis comme dépendants de *être*.

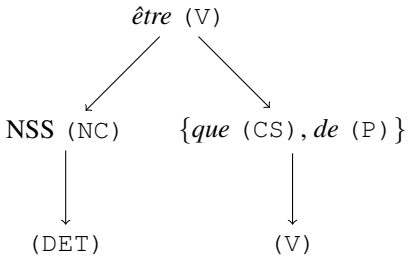


FIGURE 1 – Contextes recherchés dans les phrases du corpus

Nous donnons à la table 1 un fragment de phrase qui contient une occurrence des contextes recherchés dans le corpus.

Id.	Forme	Lemme	Ét. POS	Informations morphologiques	Id. gouv.	Fonction synt.
1	Le	le	DET	g=m|n=s|s=def	2	det
2	problème	problème	NC	g=m|n=s|s=c	4	suj
3	aujourd'_hui	aujourd'hui	ADV	_	2	mod
4	est	être	V	m=ind|n=s|p=3|t=pst	0	root
5	que	que	CS	s=s	4	obj
6	les	le	DET	n=p|s=def	7	det
7	hommes	homme	NC	g=m|n=p|s=c	9	suj
8	politiques	politique	ADJ	n=p|s=qual	7	mod
9	tentent	tenter	V	m=ind|n=p|p=3|t=pst	5	obj

TABLE 1 – Fragment de phrase analysée par Bonsaï contenant une construction spécificationnelle

2.3 Résultats de l'extraction et sélection du lexique

Au total, 28 445 phrases contiennent la structure recherchée, soit 0,48 % des phrases du corpus. Nous identifions et comptons tous les noms trouvés dans ces contextes. Au total, 1 670 noms (ou adjectifs nominalisés) différents entrent

dans les contextes recherchés dans le corpus. Nous présentons dans le tableau 2 les noms les plus fréquemment employés dans les constructions syntaxiques recherchées. Pour chaque nom (ou adjectif nominalisé), nous renseignons le nombre d'occurrences, et le pourcentage sur le nombre total d'occurrences du patron syntaxique.

Lemme	Nb. d'occurrences	%	Lemme	Nb. d'occurrences	%
objectif	3635	12.78	intérêt	365	1.28
problème	1620	5.7	priorité	324	1.14
but	1597	5.61	important	317	1.11
question	1293	4.55	risque	309	1.09
idée	1176	4.13	difficulté	288	1.01
rôle	554	1.95	chose	285	1.0
ambition	475	1.67	souci	275	0.97
mission	435	1.53	solution	249	0.88
essentiel	392	1.38	mérite	248	0.87
enjeu	382	1.34	vérité	245	0.86

TABLE 2 – Les 20 noms apparaissant le plus fréquemment dans les contextes recherchés

Au sein des constructions spécificationnelles identifiées, les NSS peuvent être modifiés par un adjectif, comme en (7), par un syntagme prépositionnel, comme en (8), ou une proposition relative, comme en (9). Il peuvent avoir un déterminant possessif, comme en (10). On trouve également dans les constructions spécificationnelles des adjectifs nominalisés comme *essentiel* ou *important*. On retrouve les hyperonymes mentionnés par Apothéloz, comme *chose* ou *côté*, qui fonctionnent généralement avec un modifieur, comme en (11).

7. En fait, le principal **problème** de cette présentation est que « les membres de l'ONU à qui elle s'adressait n'en ont rien à faire », commente le New Republic sur son site Internet.
8. L'**objectif** de l'entraîneur, c'est de donner au joueur de l'autonomie.
9. La **leçon** que j'en tire, c'est qu'il faut éviter ces passages à vide, qu'il faut de la constance.
10. Notre **position** est que nous n'aurions pas dû être en Irak, en premier lieu.
11. Le **côté** positif de cette motion de censure, c'est de ressouder la majorité autour du premier ministre.

Pour les expériences présentées dans la suite de cet article, nous conservons la portion du lexique pour laquelle le nombre d'occurrences de chaque nom dans les constructions spécificationnelles dépasse 0,1 % du total d'occurrences des constructions spécificationnelles en corpus (28 445). Le lexique ainsi sélectionné contient 122 noms, les moins fréquents du lexique retenu étant *possibilité* et *option*, avec chacun 29 occurrences. Parmi l'ensemble des noms ayant été identifiés dans les constructions spécificationnelles, 1 546 sont donc exclus du lexique retenu. Le lexique sélectionné est volontairement restreint pour éviter tout bruit lié au lexique. Les noms dont les fréquences sont les plus basses correspondent généralement à des erreurs d'analyse syntaxique, pour des phrases dont la structure est moins fréquente — voir par exemple *ljubljanais* en (12).

12. D'ailleurs, le dernier chic culinaire **ljubljanais** est de mettre sur la carte des restaurants un « simple poisson avec son filet d'huile d'olive » après une tapenade – qu'on accompagne d'un tokay de Goriska Brda, cette région proche de l'Italie qui produit de bons crus.

3 Fouille de données pour la découverte de contextes syntaxiques spécifiques

Comme nous le verrons à la section 4, nous souhaitons identifier les occurrences de NSS dans le corpus décrit à la section 2.1, afin d'extraire des séquences organisationnelles fréquentes. A priori, nous n'avons aucun moyen de distinguer dans le corpus les emplois comme NSS des items de notre lexique de leurs autres emplois sans procéder à une annotation manuelle — excepté dans les cas où ils entrent dans une construction spécificationnelle. Or, l'identification de séquences organisationnelles fréquentes suppose de travailler sur un corpus conséquent, pour lequel une annotation complètement manuelle des emplois de NSS n'est pas envisageable. Nous verrons à la section 4 que considérer toutes les occurrences des noms du lexique comme des occurrences de NSS n'est pas une solution satisfaisante en ce qui concerne le bruit, et que se limiter aux seules constructions spécificationnelles n'est pas satisfaisant non plus en ce qui concerne le silence. Nous avons donc opté pour une solution intermédiaire d'identification des NSS, pour laquelle nous ne disposions que d'un lexique d'items ayant la capacité d'être des NSS, et d'un corpus conséquent analysé pour les dépendances

syntaxiques. Cependant, nous savons que les NSS sont susceptibles d'apparaître dans des constructions syntaxiques particulières, dans lesquelles d'autres noms ne peuvent pas entrer. Pour l'anglais, des études sur corpus des emplois de NSS ont déjà été effectuées par Schmid (2000) et Flowerdew (2003), et recensent des contextes syntaxiques autres que le patron *Det N be* (*that–clause* | *wh–clause* | *to–inf*), comme le patron *Det N* (*that–clause* | *wh–clause* | *to–inf*) (Schmid, 2000) ou le patron *Det N be Det N*, dans lequel le second nom est un déverbal (Flowerdew, 2003). Pour le français, les constructions spécifiques aux NSS n'ont pas été inventoriées. On suppose qu'il existe en français d'autres constructions que la construction spécificationnelle [4] dans lesquelles on trouve des emplois de NSS. L'objectif du travail présenté dans cette section est précisément **d'identifier ces constructions**.

Nous partons de l'hypothèse que les noms communs n'appartenant pas au lexique des NSS ne peuvent pas entrer dans les constructions spécifiques aux emplois de NSS. L'idée qui sous-tend cette approche est que les propriétés discursives ou sémantiques des emplois de NSS sont corrélées à des traits syntaxiques. Pour identifier des constructions syntaxiques caractéristiques des emplois de NSS, nous utilisons une méthode permettant d'identifier des patrons linguistiques fréquents en corpus, héritée des techniques de fouille de données, et proposée par Béchet *et al.* (2012) : l'extraction de motifs séquentiels. La *fouille de données séquentielles*, introduite par Agrawal et Srikant (1995), est une technique qui permet de découvrir de nouvelles connaissances sous forme de régularités dans des bases de données, en tenant compte de l'ordre temporel entre les données. Suivant une méthodologie proche de celle de Quiniou *et al.* (2012) concernant la stylistique, nous nous intéressons aux particularités des contextes syntaxiques des NSS, via le calcul de *motifs séquentiels émergents*. Ces motifs émergents sont identifiés à partir de motifs séquentiels impliquant des noms retenus dans le lexique de NSS (étiquetés `NSS`), et de motifs impliquant des noms communs (étiquetés `NC`) — et ne contenant aucun des noms retenus dans le lexique. L'avantage de cette méthode de fouille est d'être symbolique. Les motifs extraits, et découverts automatiquement, sont interprétables par un humain. Ils se prêtent donc facilement à une analyse linguistique.

À la section 3.1, nous présentons les principes généraux du calcul de motifs séquentiels et de l'identification de motifs émergents. À la section 3.2, nous décrivons l'extraction de motifs émergents pour les NSS, et donnons les résultats quantitatifs de cette extraction. À la section 3.3, nous décrivons les principaux patrons syntaxiques caractéristiques des emplois de NSS, identifiés par l'analyse des motifs émergents.

3.1 Les motifs séquentiels et les motifs émergents

Nous définissons dans cette section les notions impliquées dans le calcul de motifs séquentiels et des motifs émergents, en les illustrant à travers des exemples similaires aux données que nous traitons ici. Néanmoins, il faut noter que la méthode de fouille de données séquentielles décrite ici trouve de très diverses applications, comme par exemple l'extraction de chaînes d'ADN ou de séquences de protéines.

Un *itemset* I est un ensemble de littéraux appelés *items*, représenté par $I = (i_1 \dots i_n)$. Par exemple, (*problème* NC) est un itemset contenant un item correspondant à un lemme et un item correspondant à une étiquette POS. Une *séquence* S est une liste ordonnée d'itemsets, représentée par $S = \langle I_1 \dots I_m \rangle$. Notons que de nombreuses applications ne nécessitent qu'un seul item dans leurs itemsets. Ces séquences sont appelées des *séquences d'items* et sont représentées par $S = \langle i_1 \dots i_n \rangle$, où $i_1 \dots i_n$ sont des items. Dans la suite de cet article, les deux types de séquences seront considérés : les séquences d'items (lemmes) et les séquences d'itemsets (lemmes et étiquettes POS). Une séquence $S = \langle I_1 \dots I_n \rangle$ est *contenue* dans une séquence $S' = \langle I'_1 \dots I'_m \rangle$ s'il existe des entiers $1 \leq j_1 < \dots < j_n \leq m$ tels que $I_1 \subseteq I'_{j_1}$, ..., $I_n \subseteq I'_{j_n}$. La séquence S est appelée *sous-séquence* de S', ce qui est noté $S \preceq S'$. Par exemple, la séquence $\langle$(`DET`) (`NC`) (*être* `V`)$\rangle$ est contenue dans la séquence $\langle$(*le* `DET`) (*solution* `NC`) (*être* `V`) (*de* `P`)$\rangle$. Une base de séquences B est un ensemble de tuples (S_{id}, S), où S_{id} est un identifiant de séquence et S une séquence. La table 3 représente ainsi une base de trois séquences. Dans notre cas, une séquence correspond à la représentation d'une phrase du corpus. Les itemsets peuvent contenir des lemmes ou des étiquettes POS.

Identifiant	Séquence
1	$\langle$(*le* `DET`) (*solution* `NC`) (*être* `V`)(*de* `P`) (*partir* `VINF`) (. `PONCT`)$\rangle$
2	$\langle$(*le* `DET`) (*problème* `NC`) (: `PONCT`)(*trouver* `VINF`) (*un* `DET`) (*financement* `NC`) (. `PONCT`)$\rangle$
3	$\langle$(*le* `DET`) (*solution* `NC`) (: `PONCT`) (*partir* `VINF`) (. `PONCT`)$\rangle$

TABLE 3 – Une base de séquences

4. (*Det N* (∅ | *ce*) *être* (*que–clause* | *de–inf*) décrite en section 2.2.

Un tuple (S'_{id}, S') *contient* une séquence S si $S \preceq S'$. Le *support* d'une séquence S dans une base de séquences B, noté *sup*(S), est le nombre de tuples contenant S dans la base. Par exemple, dans la table 3, le support de la séquence $(\langle(\textit{le}\ \texttt{DET})\ (\textit{solution}\ \texttt{NC})\rangle)$ est 2. Le *support relatif* peut aussi être utilisé, comme défini par l'équation suivante.

$$sup(S) = \frac{|\{(S'_{id}, S') \mid (S'_{id}, S') \in B \wedge (S \preceq S')\}|}{|B|}$$

Un *motif fréquent* est une séquence dont le support est supérieur ou égal à un seuil fixé : *minsup*. Les algorithmes de fouille de motifs séquentiels extraient tous les motifs fréquents apparaissant dans une base de séquences. L'ensemble des motifs fréquents pouvant être très grand, il existe une représentation condensée permettant d'éliminer les redondances sans perte d'information : les *motifs clos* (Yan *et al.*, 2003). Un motif fréquent S est clos s'il n'existe aucun motif fréquent S' tel que $S \preceq S'$ et *sup*$(S) =$ *sup*(S'). Certaines contraintes peuvent être définies pour diriger l'extraction de motifs selon les besoins de l'utilisateur et éliminer des motifs non pertinents (Dong et Pei, 2007), comme la contrainte de fréquence (en donnant une valeur au support minimal) ou la contrainte *gap* : un motif avec une contrainte *gap* égale à $[x, y]$, noté $P_{[x,y]}$, est un motif dont chaque couple d'itemsets est séparé par au moins $x - 1$ itemsets et au plus $y - 1$ itemsets.

Les motifs émergents sont des motifs dont le support augmente de manière significative d'un ensemble de données à un autre (Dong et Li, 1999). Les motifs émergents sont ainsi des motifs dont le taux de croissance ("growth rate"), c'est-à-dire le rapport des supports dans deux ensembles de données, est supérieur à un seuil fixé ρ. Un motif P d'un ensemble de données D_1 est alors un *motif émergent*, par rapport à un autre ensemble de données D_2, si $TC(P) \geq \rho$, avec $\rho > 1$ et $TC(P)$ défini par l'équation suivante.

$$TC(P) = \begin{cases} \infty & \text{si } sup_{D_2}(P) = 0 \\ \frac{sup_{D_1}(P)}{sup_{D_2}(P)} & \text{sinon} \end{cases}$$

3.2 Extraction de motifs émergents pour les noms sous-spécifiés

Pour calculer des motifs émergents relatifs aux noms sous-spécifiés, nous constituons tout d'abord une base de séquences B_{NSS}, dans laquelle chaque séquence correspond à une phrase. Notons que sur chaque séquence nous ajoutons deux itemsets particuliers – (`INIT`) et (`END`) – qui symbolisent respectivement les débuts et fins de phrase. Pour constituer la base B_{NSS}, on extrait du corpus présenté à la section 2.1 toutes les phrases contenant au moins un nom du lexique de NSS d'une part, en remplaçant leur étiquette `NC` (nom commun) par l'étiquette `NSS` (nom sous-spécifié), ou leur étiquette `ADJ` (adjectif) par l'étiquette `ASS` (adjectif nominalisé sous-spécifié). Au total, 1 881 526 phrases du corpus contiennent au moins une occurrence d'un nom du lexique retenu (soit environ 26 % des phrases), donc $|B_{NSS}|$ vaut 1 881 526. L'ensemble de ces phrases contient 2 035 850 occurrences étiquetées comme `NSS`, et 400 099 occurrences étiquetées comme `ASS`.

Pour limiter le nombre de motifs émergents à analyser, et pour extraire des motifs émergents plus génériques, nous ne conservons dans la base de séquences que l'étiquette POS (on ne tient pas compte des lemmes), excepté pour les mots appartenant aux classes fermées (`DET`, `P`, `P+D`, `CS`, `CC`, `PROREL`), ainsi que pour les auxiliaires *être* et *avoir*. Cela nous permet d'éliminer des motifs émergents trop spécifiques, se rapportant essentiellement à un élément du lexique de NSS. En effet, nous avons mené une première extraction de motifs émergents dont les séquences contiennent des itemsets avec lemmes et étiquettes POS. L'observation des motifs émergents les plus fréquents montre qu'on extrait par cette méthode beaucoup de motifs spécifiques à un NSS en particulier. Parmi les motifs ainsi extraits, on trouve des motifs positifs, comme *[poser le NSS de]* (par exemple avec *question* ou *problème*) ou *[NSS consister à]* (par exemple avec *solution*), correspondant à des occurrences de noms sous-spécifiés, et des motifs négatifs, ne correspondant pas à des occurrences de noms sous-spécifiés. Parmi les motifs émergents les plus fréquents, on trouve par exemple le motif *projet de NC* (pour *projet de loi*) qui ne peut pas être considéré comme une occurrence du NSS *projet*. C'est également le cas pour des motifs comme *faire l'objet de*, *marché du travail*, *contrat de travail*, *point de vente*, *être sur le point de*, *mettre au point*, etc. La prise en compte de ces motifs émergents plus spécifiques peut être utile pour raffiner l'identification des NSS, mais dans un premier temps, nous voulons identifier des motifs émergents plus génériques.

Nous effectuons le choix du support *minsup* pour l'extraction des motifs dans la base B_{NSS} en fixant la contrainte suivante : les motifs extraits doivent être présents dans au moins 0,1 % du nombre d'occurrences de l'étiquette `NSS`. Ce nombre d'occurrence étant de 2 035 850, le support choisi est de 2036. Pour réduire les redondances dans les motifs extraits, nous utilisons l'extraction de motifs clos. Nous fixons la contrainte *gap* à $[1, 1]$, c'est-à-dire que les items ou itemsets des motifs extraits sont séparés par 0 items ou itemsets. La longueur minimale des motifs est fixée à 2, et la longueur maximale à 7.

Après l'extraction des motifs effectuée sur la base de séquences B_{NSS}, nous retenons deux ensembles de motifs :

- l'ensemble D_{NSS}, qui rassemble les motifs calculés à partir de B_{NSS} et contenant une étiquette NSS, comme le motif (*le* DET) (NC) (*être* V) (*que* CS) ;
- l'ensemble D_{NC}, qui rassemble les motifs calculés à partir de B_{NSS} contenant des étiquettes NC mais pas d'étiquette NSS, comme (*le* DET) (NC) (V) (ADJ).

Le tableau 4 indique le nombre de motifs clos fréquents (en valeur absolue et en pourcentage) en fonction de leur longueur et pour ces 2 ensembles.

Ensemble de motifs	Longueur 2	Longueur 3	Longueur 4	Longueur 5	Longueur 6	Longueur 7	Total
D_{NSS}	79 (5.71 %)	419 (30.3 %)	527 (38.1 %)	290 (20.97 %)	65 (4.7 %)	3 (0.22 %)	1 383 (100 %)
D_{NC}	191 (3.19 %)	1 110 (18.53 %)	2 246 (37.5 %)	1 658 (27.68 %)	629 (10.5 %)	156 (2.6 %)	5 990 (100 %)

TABLE 4 – Nombre de motifs fréquents et clos extraits

Motifs émergents	Longueur 2	Longueur 3	Longueur 4	Longueur 5	Longueur 6	Longueur 7	Total
D_{NSS}/D_{NC}	1 (0.87 %)	23 (20 %)	46 (40 %)	34 (29.57 %)	11 (9.57 %)	0 (0 %)	115 (100 %)
D_{NSS}/D_{NC} (avec $TC = \infty$)	1 (1.39 %)	11 (15.28 %)	25 (34.72 %)	26 (36.11 %)	9 (12.5 %)	0.0 (0 %)	72 (100 %)

TABLE 5 – Nombre de motifs émergents

L'ensemble de motifs dans lequel nous voulons identifier des motifs émergents est D_{NSS}, c'est-à-dire les motifs contenant des étiquettes NSS. Nous avons procédé à l'extraction de motifs émergents de D_{NSS} par rapport à D_{NC}. Pour cela, on considère chaque motif appartenant à D_{NSS}, puis on calcule son taux de croissance en recherchant le même motif dans D_{NC} ; après avoir substitué à l'étiquette NSS l'étiquette NC. Le tableau 5 donne le nombre de motifs émergents de D_{NSS} par rapport à D_{NC}, tous taux de croissance confondus puis uniquement avec les motifs ayant un taux de croissance infini.

On remarque que le calcul des émergents produit un nombre de motifs relativement faible (187) qui permet une analyse manuelle. D'autre part, parmi ces motifs émergents, 72 ont un taux de croissance infini : ce sont des patrons syntaxiques caractéristiques des NSS puisque ces constructions n'apparaissent pas avec un NC. Notons que ce sont surtout les motifs de longueur 3 à 5 qui semblent les plus pertinents. Le tableau 6 montre l'ensemble des motifs émergents de longueur 3.

Parmi l'ensemble des motifs émergents, on retrouve la construction spécificationnelle. Par exemple, dans les motifs de longueur 3, on la retrouve en (e) dans le tableau 6. On la retrouve également dans les motifs de longueur 5, sous la forme (DET *le*) (NSS) (V *être*) (P *de*) (VINF). Mais de plus, l'un des résultats intéressants est que la méthode fondée sur les motifs émergents a permis de découvrir de nouvelles constructions spécifiques aux NSS. La section suivante présente des observations en corpus à partir de ces patrons caractéristiques découverts.

3.3 Patrons identifiés à partir des motifs émergents

Dans cette section, nous présentons les principaux patrons caractéristiques des emplois de NSS que nous avons identifiés grâce à l'extraction des motifs émergents et l'exploration manuelle du corpus guidée par les motifs. Les patrons identifiés correspondent à des emplois cataphoriques de NSS.

Le patron (V *avoir*) (P *pour*) (NSS) (P *de*) Ce patron a été identifié à partir de motifs émergents dont le taux de croissance est égal à ∞ et dont le support est parmi les plus élevés. Il couvre par exemple les motifs (a) et (b) du tableau 6, (P *pour*) (NSS) (P *de*) et (V *avoir*) (P *pour*) (NSS), dont on trouve respectivement une occurrence dans les exemples (13) et (14) [5], tirés du corpus *Le Monde*. Il correspond également à des motifs émergents plus longs, comme le motif de longueur 5 : (V *avoir*) (P *pour*) (NSS) (P *de*) (VINF).

13. Le gouvernement s'est fixé *pour* ***objectif*** *de* {parvenir à 1,6 % du PIB en 2008 et à 2 % en 2010}.

5. Dans les exemples présentés dans cette section, nous notons les NSS en gras, les patrons présentés en italiques, et les entités auxquelles réfèrent les NSS entre accolades.

Motif	Taux de croissance	Support absolu	Support relatif
(a) (P *pour*) (NSS) (P *de*)	∞	7 305	0.0039
(b) (V *avoir*) (P *pour*) (NSS)	∞	4 770	0.0025
(c) (NSS) (P *de*) (CLO *cla*)	∞	3 293	0.0018
(d) (NSS) (PONCT :) (VINF)	∞	3 268	0.0017
(e) (INIT) (NSS) (PONCT :)	∞	3 164	0.0017
(f) (V *avoir*) (NSS) (P *à*)	∞	3 139	0.0017
(g) (VPP *avoir*) (DET *le*) (NSS)	∞	2 868	0.0015
(h) (NSS) (V *être*) (CS *que*)	∞	2 654	0.0014
(i) (V *avoir*) (DET *aucun*) (NSS)	∞	2 626	0.0014
(j) (DET *aucun*) (NSS) (P *de*)	∞	2 614	0.0014
(k) (NSS) (P *de*) (VINF *avoir*)	∞	2 373	0.0013
(l) (NSS) (P *de*) (VINF *être*)	3.05	6 858	0.0036
(m) (DET *le*) (NSS) (CS *que*)	2.43	17 055	0.0091
(n) (NSS) (V *être*) (P *de*)	2.25	11 442	0.0061
(o) (V *avoir*) (DET *le*) (NSS)	2.02	11 512	0.0061
(p) (NSS) (P *de*) (VINF)	2.01	78 905	0.0419
(q) (P *pour*) (DET *de*) (NSS)	1.82	6 316	0.0034
(r) (DET *le*) (NSS) (PROREL *dont*)	1.65	5 894	0.0031
(s) (NSS) (P *de*) (CLR *se*)	1.6	6 768	0.0036
(t) (V *avoir*) (DET *de*) (NSS)	1.49	3 503	0.0019
(u) (V *être*) (DET *son*) (NSS)	1.19	2 584	0.0014
(v) (V *être*) (ADV) (NSS)	1.16	3 468	0.0018
(w) (DET *un*) (NSS) (PONCT :)	1.08	3 646	0.0019

TABLE 6 – Exemple de motifs émergents de longueur 3 de D_{NSS} par rapport à D_{NC}

14. Ces deux procédures *ont pour* ***résultat*** *de* {déplacer insensiblement le centre d'intérêt, tenu dans la première version par l'éblouissante héroïne, vers le personnage effacé et hypocondriaque de son mari}...

Le patron (**NSS**) (**PONCT** :) Toujours parmi les motifs émergents dont le taux de croissance est égal à ∞ et dont le support est parmi les plus élevés, on trouve des motifs comme ceux étiquetés (d) et (e) dans le tableau 6. Dans le second motif, INIT symbolise le début de phrase. On trouve respectivement des occurrences de ces motifs dans les exemples (15) et (16).

15. Leur ***objectif*** *:* {*être* identifiés comme des « adultes disponibles » avec lesquels on peut parler de tout et de rien}.
16. ***Conclusion*** *:* {à ce jour, la monnaie unique n'a guère enrayé le malaise économique européen et l'on ne peut manquer de s'interroger sur son éventuelle responsabilité dans les difficultés économiques actuelles de la zone euro}.

Les patrons (**NSS**) (**P** ***de***) (**VINF**) **et** (**NSS**) (**CS** ***que***) Parmi les motifs émergents dont le taux de croissance n'est pas égal à ∞, on trouve des motifs correspondant aux cas dans lesquels les NSS sont suivis d'un infinitif ou d'une complétive, comme les motifs (k), (l), (m) et (p). Ces motifs semblent caractériser des emplois sous-spécifiés de noms, comme le montrent les exemples en (17) et (18).

17. C'est un site commercial qui a été lancé en juin 2002, à New York, avec l'***objectif*** *de* {*mettre* les gens en relation les uns avec les autres autour d'un sujet d'intérêt commun}.
18. *L'**idée*** {*que* les inspecteurs puissent renifler les armes et les documents qui s'y rapportent sans l'aide des autorités irakiennes} est absurde.

4 Identification des occurrences de noms sous-spécifiés

Dans cette section, nous présentons la méthode utilisée pour l'identification des occurrences de noms sous-spécifiés dans le corpus de travail. Cette identification s'appuie à la fois sur le lexique sélectionné à la section 2 et les patrons identifiés à la section précédente. Nous avons procédé à une évaluation manuelle de l'identification des occurrences de noms sous-spécifiés, que nous présentons également.

Contextes recherchés lors de l'identification Lors de l'identification des noms sous-spécifiés, seules les occurrences de noms appartenant au lexique sélectionné à la section 2 sont considérées par le système. Chaque occurrence rencontrée est étiquetée comme nom sous-spécifié (NSS) lorsque son contexte d'apparition correspond à un des 6 patrons présentés ci-dessous (et dans le tableau 8), soit comme nom commun (NC) dans le cas contraire. Parmi les contextes menant à un étiquetage comme NSS, on retrouve les patrons syntaxiques décrits à la section 3.3, que nous noterons ici NSS_etre_que_de (pour la construction spécificationnelle), pour_NSS_de, NSS_punct (pour les cas dans lesquels le nom est suivi de deux points) et NSS_que_de (pour les noms prédicatifs suivis d'une complétive ou d'une infinitive). Au cours des différentes étapes du travail et de l'exploration manuelle du corpus, nous avons également identifié des contextes d'occurrences anaphoriques de NSS, qui n'ont pas été identifiées parmi les motifs émergents, et que nous avons intégrés aux contextes déclenchant l'étiquetage comme NSS. Ces contextes sont désignés par les patrons : dem_NSS, pour les occurrences dans lesquelles le NSS a un déterminant démonstratif, comme dans *ce problème* ; root_NSS, pour les occurrences dans lesquelles le NSS est la tête d'une phrase averbale, et dans lesquelles il peut être modifié par une relative, comme en (19).

19. Une **perspective** qui incite les français à accélérer leur internationalisation.

Pour chaque patron, le repérage peut être effectué de façon surfacique (c'est-à-dire en s'appuyant uniquement sur la séquences de lemmes et d'étiquettes POS, et en n'autorisant aucune distance entre les différents éléments du patron) ou en s'appuyant sur les dépendances syntaxiques, comme nous l'avons présenté à la section 2.2 pour l'identification des constructions spécificationnelles. Cela nous permet notamment de prendre en compte les cas dans lesquels le NSS est modifié par un adjectif, un groupe prépositionnel ou une relative, ce qui n'est pas permis par le repérage surfacique.

Évaluation de l'identification L'évaluation porte sur les occurrences de noms (ou adjectifs) du lexique de NSS, étiquetées préalablement par le système présenté précédemment [6]. L'annotation a été effectuée par un annotateur sur 38 documents du corpus, dans lesquels 600 occurrences de noms du lexique ont été repérées par le système, et 120 ont été étiquetées comme NSS (20 % du total des occurrences). Pour chaque occurrence, l'annotation a consisté à vérifier que l'étiquette attribuée était correcte. Pour les cas dans lesquels le rôle de nom sous-spécifié est incertain, l'occurrence a été annotée comme unknown [7].

Nous présentons dans le tableau 7 la répartition des occurrences en fonction de l'étiquetage du système et de l'annotation manuelle. Parmi les 600 occurrences de noms du lexique, 150 ont été annotées manuellement comme NSS (soit 25 %). Le rappel du système est de 0,61. En mettant de côté les occurrences étiquetées comme NSS et annotées comme unknown, la précision de l'identification des NSS effectuée par le système est de 0,83 et le F-score de 0,71. En considérant les occurrences étiquetées comme NSS et annotées comme unknown comme des faux positifs, la précision est de 0,77 et le F-score de 0,68. Une des conclusions que l'on peut tirer de cette évaluation, c'est que considérer toutes les occurrences des noms du lexique comme des occurrences de NSS est susceptible de bruiter considérablement les données extraites. Si notre système avait étiqueté toutes les occurrences de noms du lexique retenu comme NSS, la précision aurait été de 0,29, le rappel de 1, et le F-score de 0,45.

Occurrences	Annotées NSS	Annotées NC	Annotées unknown
Étiquetées NSS	92	18	10
Étiquetées NC	58	355	67

TABLE 7 – Répartition des occurrences évaluées, en fonction de l'étiquetage du système et de l'annotation manuelle

Dans le tableau 8, nous présentons le nombre d'occurrences et le pourcentage d'apparition des différents patrons sur l'ensemble des 120 occurrences étiquetées comme NSS par le système. Nous présentons également, pour chaque patron, la répartition des annotations et la précision de l'étiquetage du système (en ne comptant comme correctes que les occurrences annotées NSS). Ces résultats montrent que l'intégration d'autres patrons que les constructions spécificationnelles est nécessaire à une extraction moins silencieuse. En effet, on peut comparer les résultats de l'identification à ceux d'une identification qui s'appuierait uniquement sur la présence des constructions spécificationnelles. Nous observons que seulement 10 occurrences de NSS ont été extraites grâce à la présence du patron NSS_etre_que_de. Si l'on avait choisi de considérer comme NSS uniquement ces occurrences, le rappel du système aurait été de 0,07, la précision de 1, et le F-score de 0,13.

6. Nous n'évaluons pas la couverture du lexique sélectionné. Comme nous l'avons déjà dit, le lexique sélectionné est volontairement restreint pour interdire le bruit qui pourrait être lié au lexique.

7. Parmi les occurrences annotées unknown, on trouve essentiellement des cas dans lesquels le NSS réfère bien à une entité abstraite, mais dans lesquels cette entité est nominalisée, comme dans *« D'abord, un sentiment de fragilisation professionnelle. »*, et pour lesquels une étude plus approfondie des NSS nous permettrait de trancher.

Patron	Nb. d'occ.	Annotées NSS	Annotées NC	Annotées unknown	Précision
NSS_que_de	40 (43,5 %)	31	7	2	0,78
dem_NSS	35 (38 %)	29	2	4	0,83
NSS_punct	17 (18,5 %)	11	4	2	0,65
root_NSS	16 (17,4 %)	9	5	2	0,57
NSS_etre_que_de	10 (10,9 %)	10	0	0	1
pour_NSS_de	0 (0 %)	0	0	0	–

TABLE 8 – Proportion des différents patrons dans l'ensemble des occurrences étiquetées comme NSS et résultats de l'annotation

Pour les 58 occurrences NSS qui n'ont pas été repérés par le système, nous avons annoté le contexte d'apparition. Pour 18 d'entre eux, le contexte correspond à un patron (NSS_etre_que_de, NSS_punct ou NSS_que_de) qui n'a pas été correctement identifié, du fait d'un cas particulier dans l'analyse syntaxique. On trouve également une vingtaine d'occurrences anaphoriques et cataphoriques avec déterminant défini ou indéfini, comme dans *« La question est posée. »* (anaphorique) ou *« Revenons sur les faits. »* (cataphorique).

5 Conclusion et perspectives

Dans cet article, nous nous sommes intéressés à une classe d'items peu étudiée dans le cadre du Traitement Automatique des Langues et dans le cadre de l'analyse du discours : les noms sous-spécifiés. Nous avons essentiellement détaillé les étapes d'une identification satisfaisante des occurrences de ces noms en corpus, à savoir : l'acquisition d'un lexique d'items pouvant être des NSS dans certains de leurs emplois ; l'identification de constructions syntaxiques dans lesquelles ces items sont susceptibles d'avoir un emploi en tant que NSS. Cette identification s'appuie sur une méthode de fouille de données : l'extraction de motifs séquentiels émergents. Ces motifs émergents, constituant des patrons syntaxiques, peuvent ensuite être analysés et validés manuellement. Ces premières étapes nous ont permis (notamment par l'intégration des patrons syntaxiques identifiés à l'aide des motifs émergents) d'obtenir un système d'identification automatique des NSS en corpus, que nous avons évalué sur 600 occurrences de noms du lexique. Le résultat de cette évaluation est encourageant, puisque la précision de l'identification se situe aux alentours de 0,8, le rappel est de 0,61, et le F-score aux alentours de 0,7.

Les NSS, et plus précisément les séquences de NSS, sont des signaux potentiels de ce que nous appelons des *séquences organisationnelles*, telles que *problème–solution*. Notre objectif est maintenant d'utiliser notre système d'identification des NSS pour extraire des séquences organisationnelles fréquentes en corpus, afin d'effectuer une analyse linguistique de ces séquences, d'étudier les interactions entre ces séquences et les relations de discours, de clarifier le rôle des NSS dans l'organisation discursive, ainsi que leur statut au sein des indices de la structure discursive. L'identification automatique des séquences organisationnelles, qui peuvent a priori elles-mêmes être des signaux de relations de cohérence, pourrait permettre d'améliorer certaines tâches liées à l'analyse automatique du discours, l'extraction d'information, etc. L'extraction des séquences fréquentes devra tenir compte d'informations que nous avons jusqu'ici ignorées : les informations sur la position des noms au sein du document, au sein du paragraphe, etc. Les connecteurs discursifs seront également intégrés aux séquences organisationnelles.

Remerciements

Ce travail a bénéficié d'une aide de l'État gérée par l'Agence Nationale de la Recherche au titre du programme Investissements d'Avenir portant la référence ANR-10-LABX-0083 et du projet Hybride ANR-11-BS02-002.

Références

AGRAWAL, R. et SRIKANT, R. (1995). Mining Sequential Patterns. *In Proceedings of the Eleventh International Conference on Data Engineering*, ICDE '95, pages 3–14, Washington, DC, USA. IEEE Computer Society.

APOTHÉLOZ, D. (2008). À l'interface du système linguistique et du discours : l'exemple des constructions identificatives (e.g. pseudo-clivées). *Discours, diachronie, stylistique du français. Études en hommage à Bernard Combettes*, pages 75–92.

ASHER, N. (1993). *Reference to Abstract Objects in Discourse*. Kluwer.

ASHER, N. et LASCARIDES, A. (2003). *Logics of Conversation*. Cambridge University Press.

BÉCHET, N., CELLIER, P., CHARNOIS, T. et CRÉMILLEUX, B. (2012). Discovering linguistic patterns using sequence mining. *In Proceedings of Springer LNCS, 13th International Conference on Intelligent Text Processing and Computational Linguistics – CICLing'2012*, volume 1, pages 154–165.

CANDITO, M.-H., JOAKIM, N., DENIS, P. et HENESTROZA ANGUIANO, E. (2010). Benchmarking of Statistical Dependency Parsers for French. *In Proceedings of COLING'2010*, Beijing, China.

DENIS, P. et SAGOT, B. (2009). Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art pos tagging with less human effort. *In PACLIC 2009*, Hong-Kong, China.

DONG, G. et LI, J. (1999). Efficient Mining of Emerging Patterns : Discovering Trends and Differences. *In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '99, pages 43–52, New York, NY, USA. ACM.

DONG, G. et PEI, J. (2007). *Sequence Data Mining*, volume 33 de *Advances in Database Systems*. Kluwer.

FLOWERDEW, J. (2003). Signalling Nouns in Discourse. *English for Specific Purposes*, 22:329–346.

FLOWERDEW, L. (2008). *Corpus-based Analyses of the Problem-Solution Pattern*. Studies in Corpus Linguistics 29. John Benjamins, Philadelphia.

HIGGINS, F. R. (1979). *The pseudo-cleft construction in English*. Garland, New York.

KOLHATKAR, V., ZINSMEISTER, H. et HIRST, G. (2013a). Annotating Anaphoric Shell Nouns with their Antecedents. *In Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 112–121, Sofia, Bulgaria.

KOLHATKAR, V., ZINSMEISTER, H. et HIRST, G. (2013b). Interpreting Anaphoric Shell Nouns using Antecedents of Cataphoric Shell Nouns as Training Data. *In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 300–310.

LEGALLOIS, D. (2006). Quand le texte signale sa structure : la fonction textuelle d'une certaine catégorie nominale. *Corela*.

LEGALLOIS, D. (2008). Sur quelques caractéristiques des noms sous-spécifiés. *Scolia*, 23:109–127.

LEGALLOIS, D. et GRÉA, P. (2006). L'objectif de cet article est de... construction spécificationnelle et grammaire phraséologique. *In* LECOLLE, M. et LEROY, S., éditeurs : *Changement linguistique et phénomènes de fixation : figement, lexicalisation, catachrèse*, volume 46 de *Cahiers de praxématique*, pages 161–184. Montpellier : Publications Montpellier 3.

MANN, W. et THOMPSON, S. (1988). Rhetorical structure theory : Towards a functional theory of text organization. *Text*, 8:243–281.

NIVRE, J., HALL, J. et NILSSON, J. (2006). MaltParser : A Data-Driven Parser-Generator for Dependency Parsing. *In Proceedings of the fifth international conference on Language Resources and Evaluation (LREC2006)*, pages 2216–2219, Genoa, Italy.

QUINIOU, S., CELLIER, P., CHARNOIS, T. et LEGALLOIS, D. (2012). Fouille de données pour la stylistique : cas des motifs séquentiels émergents. *In Actes des Journées Internationales d'Analyse Statistique des Données Textuelles (JADT'12)*, Liège, Belgique.

SCHMID, H.-J. (2000). English Abstract Nouns As Conceptual Shells : From Corpus to Cognition. *Topics in English Linguistics*, 34.

SWALES, J. (1981). *Aspects of Article Introductions*. Birmingham : University of Aston.

UPTON, T. A. et COHEN, M. A. (2009). An approach to corpus-based discourse analysis : The move analysis as example. *Discourse Studies*, 11(5):585–605.

VERGEZ-COURET, M., BRAS, M., PRÉVOT, L., VIEU, L. et ATALLAH, C. (2011). Discourse contribution of enumerative structures involving 'pour deux raisons' (regular paper). *In* ASHER, N. et DANLOS, L., éditeurs : *Constraints in Discourse (CID), Agay, France*. INRIA.

YAN, X., HAN, J. et AFSHAR, R. (2003). CloSpan : Mining Closed Sequential Patterns in Large Datasets. *In SDM*, pages 166–177.