Université Paris Diderot (Paris 7) Sorbonne Paris Cité

École Doctorale de Sciences du Langage n° 132

Laboratoire Alpage (Université Paris Diderot - INRIA Paris Rocquencourt)

Doctorat de Linguistique Théorique, Descriptive et Automatique

Identification automatique des relations discursives implicites à partir de corpus annotés et de données brutes

Chloé BRAUD

Thèse dirigée par Laurence DANLOS et Pascal DENIS

Soutenue le 18 décembre 2015

Composition du jury

Laurence DANLOS	Université Paris Diderot (Paris 7)	Directrice
Liesbeth DEGAND	Université Catholique de Louvain	Présidente
Pascal DENIS	INRIA Lille Nord Europe	Directeur
Philippe MULLER	Université Paul Sabatier (Toulouse 3)	Rapporteur
Caroline SPORLEDER	Université de Göttingen	Rapporteur

université PARIS DIDEROT PARIS 7

Résumé

Le développement de systèmes d'analyse discursive automatique des documents est un enjeu actuel majeur en Traitement Automatique des Langues. La difficulté principale correspond à l'étape d'identification des relations (comme *Explication*, *Contraste* ...) liant les segments constituant le document. En particulier, l'identification des relations dites implicites, c'est-à-dire non marquées par un connecteur discursif (comme *mais*, *parce que* ...), est réputée difficile car elle nécessite la prise en compte d'indices variés et correspond à des difficultés particulières dans le cadre d'un système de classification automatique. Dans cette thèse, nous utilisons des données brutes pour améliorer des systèmes d'identification automatique des relations implicites.

Nous proposons d'abord d'utiliser les connecteurs pour annoter automatiquement de nouvelles données. Nous mettons en place des stratégies issues de l'adaptation de domaine qui nous permettent de gérer les différences en termes distributionnels entre données annotées automatiquement et manuellement : nous rapportons des améliorations pour des systèmes construits sur le corpus français ANNODIS et sur le corpus anglais du *Penn Discourse Treebank*. Ensuite, nous proposons d'utiliser des représentations de mots acquises à partir de données brutes, éventuellement annotées automatiquement en connecteurs, pour enrichir la représentation des données fondées sur les mots présents dans les segments à lier. Nous rapportons des améliorations sur le corpus anglais du *Penn Discourse Treebank* et montrons notamment que cette méthode permet de limiter le recours à des ressources riches, disponibles seulement pour peu de langues.

Abstract

Building discourse parsers is currently a major challenge in Natural Language Processing. The identification of the relations (such as *Explanation*, *Contrast* ...) linking spans of text in the document is the main difficulty. Especially, identifying the so-called implicit relations, that is the relations that lack a discourse connective (such as *but*, *because* ...), is known as an hard task since it requires to take into account various factors, and because it leads to specific difficulties in a classification system. In this thesis, we use raw data to improve automatic identification of implicit relations.

First, we propose to use discourse markers in order to automatically annotate new data. We use domain adaptation methods to deal with the distributional differences between automatically and manually annotated data : we report improvements for systems built on the French corpus ANNODIS and on the English corpus *Penn Discourse Treebank*. Then, we propose to use word representations built from raw data, which may be automatically annotated with discourse markers, in order to feed a representation of the data based on the words found in the spans of text to be linked. We report improvements on the English corpus *Penn Discourse Treebank*, and especially we show that this method alleviates the need for rich resources, available but for a few languages.

Remerciements

Tout d'abord, je tiens à remercier mes directeurs de thèse qui m'ont accompagnée tout au long de cette aventure. Merci à Pascal Denis d'avoir accepté d'encadrer mon travail, merci pour sa rigueur, pour toutes ces discussions que nous avons eues autour du discours et de l'apprentissage. Un grand merci pour m'avoir laissée explorer de nombreuses pistes, et pour toutes ses suggestions et ses conseils qui ont dirigé cette thèse et m'ont permis d'apprendre énormément. Merci à Laurence Danlos de m'avoir fait découvrir le domaine du discours dans ses cours puis de m'avoir suivie patiemment dans ce travail. Merci pour son soutien et sa disponibilité, merci pour toutes ces heures consacrées à discuter de toutes ces choses qui font que l'analyse discursive est une aventure périlleuse mais passionnante, un territoire où il y a encore tant à faire.

Je remercie également les autres membres de mon jury. Merci aux rapporteurs de cette thèse, Philippe Muller et Caroline Sporleder, d'avoir accepté de relire et d'évaluer mon travail, merci pour toutes leurs remarques. Merci également à Liesbeth Degand d'avoir accepté de lire cette thèse et de faire partie de mon jury. Merci à tous aussi pour leur disponibilité, étant donnés les délais serrés.

Merci à tous les membres du laboratoire Alpage, collègues devenus amis, qui ont tous contribué à faire de ces quatre années une période riche, stimulante et légèrement moins angoissante. Merci tout particulièrement aux doctorants, pour les activités studieuses et pour les moins studieuses aussi, d'abord mes chers voisins de bureau dans l'ordre chronologique : Enrique, Pierre et Corentin, tous trois toujours disponibles pour mes questions, toujours prêts à aider, mais aussi à faire des pauses, important les pauses. Et surtout merci à Corentin, qui a supporté le plus longtemps mon voisinage, qui m'a tellement appris, aidée, soutenue, et qui en plus m'a laissée l'embêter également chez lui, par téléphone, par chat, par tous les moyens que j'ai pu imaginer, Coco is perfect. Et puis, finalement, il fut mon principal compagnon de galère d'écriture, à se fendre la gueule à coup de hache, hein ? ! Merci de m'avoir supportée, dans tous les sens du terme. Merci grandement également à Emmanuel, compagnon de voyages à Lille, merci pour son aide et son soutien constant, merci pour les poissons et les pirates, pour les Worms et les bons fous rire sur pas mal de choses. Merci aussi à Charlotte, qui m'a en quelque sorte introduite dans le milieu du discours, nos discussions discursives autour d'un café-clope m'ont tellement manqué ces deux dernières années. Merci à Marianne, toujours au top, toujours le sourire, toujours réconfortante, toujours une bonne blague ou une chanson, elle fait même de l'inception de banane, et oui, j'en ris encore. Merci à Marion, pour nos années LI, pour être venue avec moi dans cette "galère", pour ces quelques journées où tu étais avec nous, on se sera bien marrées, et les conférences et les répétitions, on aura bien stressé. Merci aussi pour cette après-midi entière passée à corriger par téléphone cette thèse, infiniment ! Merci Valérie, pour son humour génial, son sens de la décoration qui a rendu notre bureau bien plus intéressant, son soutien et les services si tentants proposés. Merci aux anciens doctorants qui m'ont accueillie à mon arrivée et m'ont montré qu'il y avait bien une fin à tout ça, merci Luc, Rosa, Juliette. Merci enfin aux nouveaux doctorants, Timothée, Maximin et Sarah, et puis, bon courage !

Un grand merci aussi aux autre membres d'Alpage, qui m'ont tous aidé à un moment ou à un autre, d'une manière ou d'une autre : Djamé, Ben C, Benoît, Vanessa, Marie, Éric, Virginie, Lucie, Mathieu. Merci pour leur aide, pour leur soutien, toutes ces discussions qui m'ont beaucoup apporté et toutes ces soirées de décompression nécessaire. Un grand merci à Ben C, Djamé, Éric, mais aussi

Alexandre, Guillaume, François et les doctorants pour ces souvenirs mémorables à Dublin, Lisbonne et ailleurs, certaines images ne pourront jamais s'effacer, certains chants resteront inoubliables, mais surtout, beaucoup de conseils et de questionnements scientifiques sont et seront toujours avec moi. Djamé, merci de m'avoir permis de rencontrer ma future équipe.

Je tiens également à remercier tous ces gens qui m'ont guidée et soutenue, avant et pendant la thèse. Merci à Olivier Bonami, sans qui je ne ferais probablement pas de TAL et sans qui je ne serais probablement pas venue à Paris 7, donc sans qui cette thèse n'existerait pas. Merci de m'avoir conseillée et orientée, mais surtout un grand merci pour m'avoir fait découvrir et aimer la linguistique. Merci également à Francis Corblin pour ses cours passionnants. Merci à Pascal Amsili, pour le cursus LI, pour les cours bien sûr, mais aussi pour le soutien permanent, en tant qu'étudiante, doctorante, moniteur. Merci pour toutes ces discussions autour de la façon d'enseigner, autour d'une phrase, autour d'un quantifieur. Et merci pour cette disponibilité incroyable, merci pour les conseils, je repartais toujours plus légère de ton bureau, merci. Merci aux toulousains de l'IRIT, pour tous nos échanges, toutes ces discussions stimulantes, et merci pour Annodis ! Un grand merci à Juliette, mon autre compagnon de galère d'écriture et d'aventures discursives. Merci à Sarah, Camille, Cyril, Xavier et Véronique, que j'ai rencontré au LIMSI avant cette thèse, et qui m'ont aidée à confirmer mon goût pour la recherche.

Merci à mes amis, qui sont restés tout près de moi, malgré mon indisponibilité croissante et ma fréquente mauvaise humeur. Merci à Juliette, pour nos années LI, d'avoir été la meilleure binôme de projet imaginable, à la fois rigoureuse et perfectionniste, mais aussi drôle et déjantée. Merci à Élodie, la voisine rêvée, avec qui j'ai partagé joies et angoisses, merci pour la célébration par le crabe, merci pour les soirées de détente. Merci à Francis, malgré la rareté de mon temps libre, nos déjeuners étaient chaque fois des bouffées d'air pur, d'un réconfort infini. Merci à Marie (celle qui ne cesse de pousser Paul), de Paris à Barcelone, en passant par Berlin et Leucate, toujours présente quand c'est nécessaire et quand ça ne l'est pas (ce qui est au moins aussi important). Merci ma rebelle, merci Magrat, merci d'être là. Merci à Ève et Jérémy, bien lointains toutes ces années, mais bizarrement toujours proches. Merci de m'avoir fait rêver de paysages merveilleux, d'aventures incroyables, de plantes et animaux fantastiques, de rencontres avec des platypus et des tortues stellaires. Merci ma petite bulle d'amour, ma chère Mémé, merci d'exister.

Un merci infini à mes proches. Tout d'abord à celui qui a été à mes côtés toutes ces années, que je ne saurais suffisamment remercier, merci à Nicolas, pour son soutien indéfectible. Merci de ne pas m'avoir laissée partir faire des études de chocolatier confiseur au bout de 3 ans, merci de m'avoir fourni en nourriture et linge propre ces derniers mois, d'avoir supporté le petit zombie avec son Macbook, qui ne parle plus et qui n'entend plus. Merci d'avoir encaissé mes angoisses et supporté patiemment mes colères, merci pour toutes tes petites et grandes attentions, merci d'avoir toujours réussi à me faire rire. Merci maman, merci papa, pour tellement, et plus encore ! Pour tout depuis toujours, bien sûr, et surtout pour toutes ces années où vous avez (très) patiemment attendu que je trouve enfin ma voie et que je finisse mes études (ça y est, on y est arrivé !). Merci infiniment pour votre soutien affectif, gastronomique, culturel, psychologique, climatique, logistique, financier ... Merci à toute ma famille, les Guilleman, Braud, Andry, Bernade, Anderhuber et Touzan, merci pour votre soutien tout au long de ces années et tout au long de ma vie. Une pensée particulière pour ceux qui ne sont plus, qui restent lovés dans mon coeur, et surtout, Olivier, ta créativité, ton sourire, tes yeux remplis d'étoiles, tu m'as tellement inspirée, tu me manques.

Finalement, un grand merci à ces humains géniaux qui ont enrichi ma vie : Pratchett, Pennac, Rivais, Eco, Céline, Chevillard, Brel, Vian, Hamilton, Asimov, Bayard, Beckett, Flaubert, EA et LucasArt, les Monty Python, Wes Anderson, Rabelais, Bosch, Miyasaki, Satoshi Kon, Steam, Michaux, Blois, Dalí, Verlinde, C2C, the Do, Stromae, Tati ... et j'en oublie. Tant de gens à remercier, veuillez excuser mes probables oublis, la thèse n'est finalement pas (tant que ça) une aventure solitaire. Merci à ce Fortune Cookie qui m'a tout appris : *There's no royal way to learning*.

Table des matières

Liste des tableaux

Table des figures

1 Introduction

Sommaire

1.1 Enjeux de l'identification des relations discursives implicites

Analyser un document nécessite d'identifier les liens qui s'établissent entre les phrases et les propositions qui le constituent. Ces liens, qui peuvent être de type causal, comparatif ou narratif par exemple, sont appelés *relations discursives*, *rhétoriques* ou *de cohérence*. Les relations discursives sont généralement vues comme des prédicats binaires prenant en *arguments* deux segments textuels, ou plus précisément leur contenu abstrait. L'ensemble de ces liens forme la *structure discursive* du document, structure qui reflète sa cohérence : chaque segment, généralement de type propositionnel, est lié à une autre partie du texte et joue un rôle par rapport à l'ensemble. Ainsi le document (1) [1], issu du corpus discursif français ANNODIS (AFANTENOS et al., 2012a), peut être représenté par la structure présentée dans la figure 1.1. Nous reproduisons la *segmentation* du texte (1) proposée dans ce corpus, c'est-à-dire le découpage du texte en *unités élémentaires de discours* ou EDU (*Elementary Discourse Unit*) ici repérées par des crochets. Le graphe 1.1 représente la structure pour le document constitué des relations discursives s'établissant directement entre des paires d'EDU, comme la relation *Condition* [2] entre 3 et 4, ou entre des paires de segments déjà liés par une relation, ou CDU (*Complex Discourse Unit*), comme la relation *Background* s'établissant entre la seconde phrase du document (constituée des unités 3 et 4 liées par la relation *Condition*) et la phrase suivante (unité 5). Dans cette thèse, nous nous intéressons à la construction de systèmes automatiques pour l'identification de ces relations.

(1) [Présidente de la Délégation aux droits des femmes de l'Assemblée nationale,]$_1$ [Marie-Jo Zimmermann est inquiète.]$_2$ [Si les réformes envisagées par le gouvernement pour les européennes et les régionales sont adoptées telles qu'envisagées,]$_3$ [la parité hommes-femmes en politique risque de régresser.]$_4$ [Et ce, alors que la France est déjà fort en retard par rapport à nos voisins européens et même par rapport à de nombreux pays dans le monde.]$_5$ [Pour les européennes, (le gouvernement) prévoit huit grandes régions mais avec des sous-sections par régions administratives qui n'éliraient qu'un, deux ou trois députés.]$_6$ [Dans ce cas, les élus seraient essentiellement les têtes de liste et donc rarement des femmes.]$_7$ [Le

1. Nous avons modifié ce document pour en réduire la taille en supprimant des segments constituant des commentaires ou des élaborations d'autres segments. Le texte de l'unité 6 est également légèrement modifié, nous avons remplacé le pronom « il » par son référent, « le gouvernement ».

2. Nous utilisons les noms anglais des relations. Notons que dans le corpus français ANNODIS, chaque relation est associée à une étiquette en anglais. Il nous a donc semblé plus simple de garder toujours les noms anglais. Comme une même étiquette ne recoupe pas toujours la même définition selon les cadres théoriques et corpus, un nom de relation sera toujours à comprendre en lien avec un certain jeu de relations, ici celui du corpus ANNODIS.

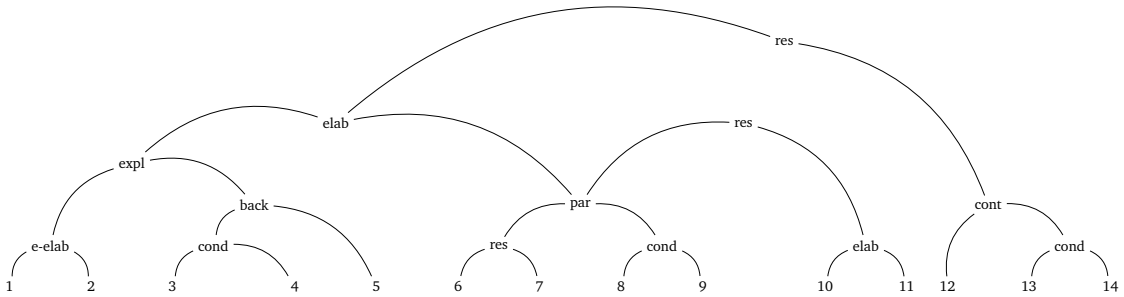


Figure 1.1.: Structure pour le document (1), les EDU sont représentées par des nombres correspondant à la segmentation du texte (1) [3]. Les relations annotées sur les arcs : « e-elab » (*Entity Elaboration*), « elab » (*Elaboration*), « expl » (*Explanation*), « cond » (*Conditional*), « back » (*Background*), « res » (*Result*), « cont » (*Continuation*), « par » (*Parallel*).

même phénomène se retrouverait aux régionales]$_8$ [si chaque liste est subdivisée en sections départementales avec une obligation de parité par tranche de six.]$_9$ [Seule solution :]$_{10}$ [appliquer une stricte alternance homme-femme dans chacune des sections.]$_{11}$ [La députée de Moselle mène donc campagne.]$_{12}$ [Et menace de saisir le Conseil constitutionnel]$_{13}$ [si les projets du gouvernement ne respectent pas l'article 3 de la Constitution.]$_{14}$

Le développement de systèmes permettant de construire automatiquement cette structure, systèmes appelés *analyseurs discursifs*, est un enjeu actuel majeur dans le domaine du Traitement Automatique des Langues comme le montre le nombre croissant de publications. Les systèmes complets existants obtiennent cependant des scores relativement bas (Muller et al., 2012a ; Ji et Eisenstein, 2014b ; Feng et Hirst, 2014 ; Li et al., 2014a ; Joty et Moschitti, 2014 ; Lin et al., 2014 ; Xue et al., 2015). Leur étude permet de conclure que l'étape d'identification de la relation liant deux segments textuels constitue la plus grande difficulté. Notamment, cette identification est particulièrement difficile dans le cas des relations dites *implicites* c'est-à-dire pour lesquelles le locuteur n'utilise pas un marqueur explicite, un *connecteur discursif*, pour signaler la relation. Ainsi, la relation *Condition* entre les unités 3 et 4 est marquée explicitement par le connecteur *si* qui exprime principalement ce type de relation et qui est obligatoire, rendant cette relation facilement identifiable ce qui se reflète dans les scores hauts obtenus pour la tâche d'identification d'une relation explicitée par un connecteur (Pitler et al., 2008 ; Pitler et Nenkova, 2009 ; Versley, 2011 ; Johannsen et Søgaard, 2013). Par contre, aucun connecteur ne signale la relation explicative entre la première phrase et les deux phrases suivantes. C'est au lecteur d'inférer à partir du contenu des propositions et de ses connaissances sur le monde que l'inquiétude de Marie-Jo Zimmermann vient du fait que la parité entre hommes et femmes risque de régresser dans le système politique, inférence liée au fait que Mme Zimmermann est une députée en charge du respect de cette parité et que cette régression peut constituer un sujet d'inquiétude. Les systèmes actuels pour l'identification automatique de ces relations implicites obtiennent des performances encore basses (Rutherford et Xue, 2014 ; Ji et Eisenstein, 2014a ; Rutherford et Xue, 2015) alors que ce type de lien correspond à environ la moitié des relations exprimées dans les textes (Soria et Ferrari, 1998 ; Stede, 2004 ; Sporleder et Lascarides, 2008 ; Taboada, 2006 ; Prasad et al., 2008a ; Subba et Di Eugenio, 2009 ; Oza et al., 2009). De plus, une erreur dans l'identification d'un lien se propage dans l'ensemble de la structure discursive construite car les relations peuvent imposer des contraintes entre elles (Lin et al., 2009 ; Roze, 2013 ; Feng et Hirst, 2014). Nous cherchons donc dans cette thèse à améliorer

3. Les relations présentes dans cette structure correspondent globalement à celles annotées dans le corpus avec quelques changements dus à la modification du document. Cette structure se veut relativement neutre de tout cadre théorique, elle repose malgré tout sur certains concepts notamment concernant les relations s'établissant entre des CDU. Ainsi, la relation *Result* qui domine le graphe s'établit de manière plus locale entre l'unité 2 et chacune des unités 12 et 13 : Mme Zimmermann est inquiète donc elle mène campagne et donc elle menace de saisir le Conseil constitutionnel. Cette interprétation correspond de manière générale à la définition d'un élément plus important au sein de chaque sous-partie du graphe qui peut remonter et constituer l'argument de relations plus hautes.

les performances sur cette tâche spécifique avec la visée de construire des analyseurs discursifs complets performants.

L'identification automatique de la structure discursive d'un document peut être utile pour différentes applications de Traitement Automatique des Langues comme les systèmes question-réponse ou le résumé automatique, et peut permettre d'améliorer d'autres tâches comme l'analyse temporelle ou la résolution de la coréférence. Ainsi, si l'on se demande en quoi les réformes envisagées par le gouvernement risquent de mettre à mal la parité au sein du système politique, on peut se fonder sur la structure discursive du document (1) telle que représentée dans le graphe 1.1 : l'élaboration du problème contenue dans les segments 6 à 9 correspond à deux exemples, mis en parallèle, de réformes envisagées. On peut également trouver la réponse apportée par la députée au problème qu'elle soulève : le résultat de son inquiétude correspond aux solutions mises en place exprimées dans les segments 12 à 14, à savoir mener campagne et menacer de saisir le Conseil constitutionnel. Pour un résumé automatique, on s'intéresse à un principe hiérarchique entre les arguments postulé dans différents cadres théoriques : selon la relation qui s'établit entre deux segments, on peut parfois considérer l'un des deux comme plus important que l'autre. On peut ainsi éventuellement supprimer l'unité 3 dans la seconde phrase, c'est-à-dire la proposition conditionnelle, tout en conservant la cohérence et l'information principale de la phrase. De même, pour raccourcir le document, nous avons supprimé des éléments élaborant ou commentant le sujet principal en nous fondant sur l'analyse discursive ou plus précisément le type de relation annoté. Quant à l'enchaînement temporel entre l'état d'inquiétude exprimé dans l'unité 2 et les actions envisagées dans les segments 12 (mener campagne) et 13 (menacer de saisir le Conseil constitutionnel), il peut être inféré par la présence d'une relation *Result*, la cause exprimée dans l'unité 2 précédant l'effet, les actions envisagées. Enfin, concernant la résolution de la coréférence, des contraintes structurelles permettent de restreindre les sites possibles d'attachement d'une anaphore. Par exemple, si une phrase commençant par le pronom « Elle » était insérée après l'unité 4, il ne serait pas possible de le rattacher à « Marie-Jo Zimmermann », il réfèrerait obligatoirement au nom commun « la parité », impossibilité directement inférable à partir de la structure. La liste des applications possibles pouvant profiter d'une analyse au niveau discursif est plus longue, des travaux ont ainsi également été menés dans le cadre de l'analyse de sentiment ou de la traduction automatique. La dérivation automatique de la structure discursive est d'un intérêt majeur pour le domaine du Traitement Automatique des Langues mais son utilisation pratique est conditionnée par l'obtention de performances suffisamment hautes, performances en grande partie dépendantes aujourd'hui de la tâche d'étiquetage en relation.

La tâche d'identification automatique des relations implicites est relativement récente mais a donné lieu à un nombre croissant de travaux ces dernières années, notamment depuis la parution du corpus anglais du *Penn Discourse Treebank* (PRASAD et al., 2008a) qui distingue, dans l'annotation, relations explicites et implicites. Cette tâche est modélisée comme un problème de classification dans le cadre d'un système d'apprentissage automatique. L'apprentissage automatique concerne le développement d'algorithmes capables de découvrir et d'exploiter des régularités dans un ensemble de données notamment pour faire des prédictions (par exemple de classe, de structure) sur de nouvelles données similaires. C'est une discipline au croisement de l'informatique et des statistiques qui connaît depuis plusieurs années un succès grandissant dans divers domaines dont le Traitement Automatique des Langues, notamment grâce au fait que de plus en plus de données sont disponibles. La classification est l'un des nombreux problèmes étudiés en apprentissage automatique : le but de l'algorithme est d'attribuer à un objet (comme une paire de segments textuels) une valeur discrète parmi un ensemble fini de valeurs représentant des classes d'objets (par exemple, une relation

discursive) [4]. Notre tâche a été modélisée à partir de classifieurs binaires, prédisant une étiquette parmi deux classes, ou de classifieurs multiclasses, le nombre de classes est alors supérieur à deux ce qui est considéré comme un cas plus difficile. De plus, certaines relations sont représentées par moins d'exemples que d'autres, problème connu sous le nom de *déséquilibre des classes*.

Un algorithme d'apprentissage automatique prend en entrée des objets, que l'on appellera *instances*, en général des vecteurs. La façon dont on transforme l'objet de départ (ou *observation*), par exemple les arguments d'une relation de discours, en une instance, donc un vecteur, correspond au choix de la représentation, c'est le problème de la modélisation des données. Les dimensions des vecteurs en entrée sont généralement appelées des *traits*, ils correspondent à des caractéristiques, des attributs des données. Le choix d'une représentation est difficile pour notre tâche, car l'inférence d'une relation discursive repose sur une variété d'indices. De plus, certaines modélisations proposées engendrent des vecteurs de très haute dimensionnalité, menant au problème dit d'*éparpillement des données* (*data sparsity*) : plus le nombre de traits est élevé, plus la quantité d'instances d'entraînement nécessaires est importante, puisque le système doit rencontrer chaque configuration, les traits et leurs valeurs, suffisamment souvent pour construire une généralisation, apprendre une régularité pour ces dimensions. Si le nombre de dimensions est très élevé par rapport au nombre d'instances disponibles, on dira que les instances sont éparpillées, ce qui conduit généralement à des systèmes peu performants, car il est difficile de construire une bonne généralisation, et éventuellement au problème dit de *sur-entraînement*, le système apprenant des associations trop spécifiques.

Enfin, notons qu'il existe différentes familles d'approches en apprentissage. Notamment, on peut distinguer l'apprentissage *supervisé* — où l'on dispose d'un ensemble de données, dites *données d'entraînement*, annotées, donc associant des objets à des étiquettes, comme des classes —, et l'apprentissage *non supervisé* — où l'on ne dispose que de données non annotées. Entre ces deux extrêmes, l'apprentissage *semi-supervisé correspond* aux cas où l'on dispose de données non annotées et de données annotées (ou d'informations sur les étiquettes). La tâche d'identification des relations discursives implicites s'instancie en général dans un cadre supervisé, mais il a également été proposé d'utiliser des instances d'entraînement annotées automatiquement, cadre de l'apprentissage *distant* (Mintz et al., 2009) ou *faiblement supervisé* (Craven et Kumlien, 1999) pour lequel au moins une partie des données correspond à des étiquettes peu sûres ou bruitées (i.e. contenant des erreurs).

Les directions de recherche possibles pour l'amélioration de la tâche d'identification des relations discursives implicites sont nombreuses. Dans cette thèse, nous nous sommes donnée comme principal défi de développer des stratégies qui ne font appel qu'à un minimum de supervision. Il est important aujourd'hui de s'interroger sur la possibilité d'améliorer des systèmes de Traitement Automatique des Langues sans rendre nécessaire l'annotation de nouvelles données ou l'utilisation de ressources construites à la main. En effet, le recours à ces stratégies nécessite un travail long et coûteux qu'il ne sera donc probablement pas possible de mener avec la même richesse pour toutes les langues. Ceci est particulièrement vrai pour le domaine discursif pour lequel l'annotation est reconnue particulièrement difficile et qui rend nécessaire l'apport d'informations très variées. Cette problématique nous a conduit à étudier différentes approches reposant sur l'apport de données brutes. Nous décrivons dans la section suivante ces approches en identifiant les difficultés liées à la tâche auxquelles elles entendent répondre ainsi que les problématiques qu'elles soulèvent.

4. Il existe de nombreux autres problèmes d'apprentissage (voir par exemple (Mohri et al., 2012 ; Hastie et al., 2001)) comme la régression, le but étant alors de prédire une valeur réelle (par exemple, le prix d'une maison), l'identification d'une distribution (*density estimation*) ou l'ordonnancement d'objets (*ranking*) etc. . .

1.2 Utilisation de données brutes pour l'identification des relations discursives implicites

Dans cette thèse, nous nous posons les questions suivantes : Comment peut-on apprendre une tâche de classification en utilisant des données brutes ? Quels sont les nouveaux problèmes introduits par l'utilisation de données brutes et comment peut-on les gérer ? L'apport de ces données brutes entraîne-t-il des améliorations en termes de performance des systèmes et, si oui, sous quelles conditions ? Nous avons choisi d'instancier ces problématiques à travers la tâche d'identification des relations discursives implicites car c'est une tâche cruciale qui présente des enjeux et des difficultés qui rendent particulièrement pertinente la mise en place d'une telle configuration.

1.2.1 Des données brutes pour augmenter l'ensemble d'entraînement

Il existe aujourd'hui plusieurs corpus annotés au niveau discursif pour différentes langues dont l'anglais et le français, corpus cependant de taille limitée. Nous pensons que l'un des obstacles actuels à l'amélioration des performances pour notre tâche réside dans cette limitation car un nombre important de données est nécessaire pour construire un système de classification automatique performant et robuste pour un problème aussi complexe, au sens que nous avons évoqué, c'est-à-dire dont la modélisation nécessite la prise en compte de nombreuses informations. De plus, certaines relations sont représentées par un nombre très restreint d'instances, ceci étant dû à la distribution des relations (leur fréquence) au sein du corpus annoté : nous ne pouvons pas ignorer ces relations, d'abord parce que cela correspondrait à une erreur dans un document donc dans l'intégralité de la structure discursive qui serait construite, mais aussi parce qu'il est possible qu'elles soient plus fréquentes dans un autre genre de données. Comme annoter manuellement de nouvelles données ne nous semble pas envisageable, la solution que nous proposons ici est d'utiliser des données supplémentaires obtenues à partir de corpus bruts pour augmenter les données d'entraînement. Il est bien sûr nécessaire que ces données aient un lien avec la tâche de classification des relations discursives. La solution la plus évidente dans ce cadre correspond à l'apprentissage semi-supervisé, donc à l'ajout de données brutes non annotées. Mais obtenir des données brutes non annotées de relations implicites est difficile car les instances ne correspondent pas à une délimitation graphique fixée comme c'est le cas, au moins partiellement, pour les instances en analyse morpho-syntaxique ou syntaxique. Il est par contre relativement facile d'obtenir des données explicites qui correspondent, comme les implicites, à des paires de segments textuels liés par des relations discursives. Nous explorons donc dans cette thèse la possibilité d'utilisation de données explicites pour améliorer l'identification des relations implicites, une ligne de recherche initiée par Marcu et Echihabi (2002).

Les données explicites peuvent être obtenues soit en utilisant les données explicites manuellement annotées dans les corpus soit en annotant automatiquement des textes bruts avec ce type de relation en utilisant des heuristiques et/ou des modèles. La première option est la plus simple mais elle nous confronte également au problème de limitation dans la taille des données disponibles, les données explicites annotées manuellement étant globalement aussi nombreuses que les implicites. Dans ce cas, on est, du moins en apparence, dans un cadre classique d'apprentissage supervisé, observations et étiquettes étant manuellement annotées. La seconde option nous permet d'obtenir un nombre quasiment illimité de nouvelles instances mais ces données créées peuvent contenir du bruit dû aux méthodes utilisées pour les acquérir, ce qui nous ramène au cadre de l'apprentissage faiblement supervisé ou distant présenté précédemment. Quelle que soit l'option choisie pour obtenir les données supplémentaires, utiliser des données explicites pour construire un système d'identification des relations implicites n'est pas anodin. Nous avons vu que ces deux types de

relation présentaient une différence importante : le signalement ou non de la relation par une marque forte, un connecteur discursif. Afin de forcer la ressemblance entre ces deux types de données, nous supprimons le connecteur discursif dans les données explicites et aboutissons ainsi à ce que nous appelons des relations implicites *artificielles*, c'est-à-dire des instances de relations implicites créées, non attestées, par opposition aux relations implicites manuellement annotées donc *naturelles*.

L'utilisation de données artificielles pour la tâche d'identification des relations implicites est une stratégie qui a d'abord été proposée par MARCU et ECHIHABI (2002) puis reprise sans grand succès dans (BLAIR-GOLDENSOHN et al., 2007 ; SPORLEDER et LASCARIDES, 2008 ; PITLER et al., 2009). Ce manque de succès est dû selon nous au fait que la construction du modèle doit prendre en compte le fait que les données artificielles et naturelles présentent de nombreuses différences, ce qui n'était pas le cas dans ces études. En effet, nous n'avons aucune garantie que ces deux types de données mettent en jeu les mêmes indices ou expriment les relations discursives avec la même fréquence. De plus, la suppression du connecteur peut avoir un effet sur l'observation, la rendant agrammaticale, incohérente ou modifiant la relation inférable. Enfin, l'utilisation d'heuristiques pour construire les données artificielles introduit du bruit évidemment absent des données naturelles. Nous avons donc deux ensembles de données similaires, liées à une même tâche, mais qui peuvent comporter des différences en termes d'entrée du système, les paires de segments, de sortie, les relations, ou du lien entre entrée et sortie. Chacun de ces éléments — instance, classe, paire (instance,classe) — peut être vu comme une variable aléatoire tirée d'une certaine distribution sous-jacente inconnue. Une hypothèse fondamentale qui sous-tend un système d'apprentissage automatique est que le système est construit et évalué à partir de données tirées indépendamment et identiquement de la même distribution : chacune des variables résulte donc d'un tirage aléatoire suivant une même loi de probabilité inconnue. C'est l'hypothèse *i.i.d.* (*independently and identically distributed*). Nous envisageons dans cette thèse les différences entre chacun des ensembles de données en termes distributionnels : la construction d'un modèle statistique à partir des données naturelles et artificielles correspond à une configuration d'apprentissage avec des *données non identiquement distribuées*, configuration qui va donc à l'encontre de l'hypothèse i.i.d., ce qui explique vraisemblablement les basses performances obtenues précédemment.

Certaines études sont parvenues à obtenir des améliorations en prenant en compte cette différence à travers une sélection des exemples tout en restant dans un cadre supervisé classique, donc en considérant l'hypothèse i.i.d. respecté pour les données utilisées (WANG et al., 2012 ; RUTHERFORD et XUE, 2015) ou en se fondant sur des algorithmes d'apprentissage multitâche qui permettent, d'une certaine manière, de gérer cette non identité distributionnelle (LAN et al., 2013). Nous choisissons ici de nous inscrire dans le cadre de l'*adaptation de domaine* (JIANG, 2008 ; PAN et YANG, 2010 ; LI, 2012 ; SØGAARD, 2013). Ce cadre a été défini pour des situations où l'on dispose d'un modèle performant sur un ensemble de données dites *sources* que l'on veut adapter à un autre ensemble de données dites *cibles* qui divergent des sources notamment en termes de genre ou de thème, plus généralement de domaine. Pour le domaine cible, on ne dispose pas ou pas suffisamment de données annotées. Les méthodes développées en adaptation de domaine ont donc pour visée de construire un système performant sur un certain ensemble de données à partir de données similaires en termes de tâche mais correspondant à des caractéristiques différentes. En considérant les données naturelles comme les données cibles et les données artificielles comme les sources, notre but est d'adapter un modèle construit sur les données artificielles aux naturelles. Nous ne nous plaçons pas exactement dans le cadre de l'adaptation de domaine au sens où, notamment, nous ne nous intéressons pas aux performances du modèle construit sur les données sources. Ce cadre nous fournit cependant des méthodes permettant de gérer les différences distributionnelles, obstacle principal au succès de cette stratégie.

1.2.2 Des données brutes pour construire une représentation

La seconde approche que nous envisageons dans cette thèse repose sur une transformation de la représentation des données. L'identification automatique des relations discursives implicites est un problème complexe en partie parce qu'elle repose sur des informations lexicales, syntaxiques, temporelles, aspectuelles mais aussi sémantiques et des connaissances sur le monde (Asher et Lascarides, 2003). En conséquence, de nombreuses études sur cette tâche ont cherché à incorporer certaines de ces informations dans leurs systèmes à travers l'utilisation d'analyseurs syntaxiques, de modèles d'identification des évènements et des entités et des liens temporels ou coréférentiels, et de lexiques de polarité, de sentiments et de catégories sémantiques (Pitler et al., 2009 ; Lin et al., 2009 ; Louis et al., 2010b ; Park et Cardie, 2012 ; Rutherford et Xue, 2014). Au contraire, les premiers systèmes pour l'identification des relations implicites, et notamment (Marcu et Echihabi, 2002), se fondaient sur une représentation surfacique correspondant aux paires de mots dans le produit cartésien sur les arguments — les segments textuels liés par la relation —, une représentation qui a généralement été reprise par la suite puis abandonnée car elle ne semblait plus nécessaire quand des ressources suffisamment riches étaient utilisées (Park et Cardie, 2012). De plus, prise seule, cette représentation des données correspond à des scores relativement bas, ce qui est attendu car elle nous confronte à un problème bien connu en apprentissage sur des données textuelles et évoqué précédemment, l'éparpillement des données, et parce qu'il est difficile d'utiliser directement cette représentation surfacique pour dégager des informations pertinentes pour une tâche qui est essentiellement sémantique. Ce sont ces deux difficultés par ailleurs liées que l'on cherche à gérer en utilisant des ressources riches et des outils d'analyse automatique fins. Cette stratégie pose cependant problème au sens où elle n'est pas applicable pour des langues moins bien dotées que l'anglais. Il nous a donc semblé intéressant de déterminer s'il était possible d'obtenir de bonnes performances en nous limitant à une représentation surfacique simple et à l'utilisation de données brutes. Notons que la stratégie reposant sur l'augmentation des données d'entraînement à travers l'utilisation d'un corpus de relations implicites artificielles constitue aussi une solution à l'éparpillement et nous évaluons donc également l'impact de cette approche sur les performances obtenues avec la même représentation surfacique.

Pour mettre en œuvre cette seconde approche, nous nous fondons sur un mouvement qui a pris de l'ampleur en particulier ces dernières années en Traitement Automatique des Langues : la construction de manière non supervisée de représentations de mots, représentations clusterisées ou « plongements » lexicaux (*word embedding*) (Brown et al., 1992 ; Collobert et Weston, 2008 ; Lebret et Collobert, 2014). Ces représentations ont été obtenues de manière non supervisée à partir de textes bruts, il est donc possible de les acquérir pour n'importe quelle langue. Elles ont conduit à des améliorations sur différentes tâches non discursives (Turian et al., 2010 ; Chen et al., 2013) ce que l'on peut expliquer par le fait qu'elles permettent de réduire l'éparpillement en conduisant à une représentation de moins grande dimensionnalité et qu'elles encodent, de par leur conception, une information paradigmatique sur les mots. En effet, le problème d'éparpillement soulevé par la représentation surfacique provient de l'encodage, dit *one-hot*, utilisé pour construire la modélisation dans le cadre d'un système statistique : on aboutit à un vecteur de très haute dimensionnalité dont seules quelques dimensions sont activées pour une instance. Les représentations de mots permettent de condenser l'information et de généraliser les observations soit en attribuant à chaque mot un code binaire correspondant au cluster auquel il appartient, défini en termes de son apparition dans certains contextes, soit en lui attribuant un vecteur à valeurs réelles dont chaque dimension peut avoir une interprétation sémantique ou syntaxique liée également à ses contextes d'apparition.

L'une de ces représentations, fondée sur un algorithme de clusterisation, a été utilisée dans une précédente étude sur les relations discursives implicites (RUTHERFORD et XUE, 2014), ces auteurs ayant montré l'impact positif de cette ressource pour la tâche. Cependant cette étude ne met pas en œuvre la comparaison, pourtant cruciale, avec un système fondé sur une représentation brute des mots. Dans cette thèse, nous mettons en compétition différents types de représentations de mots, dont celle traditionnellement utilisée reposant sur un encodage one-hot, afin d'évaluer leur capacité à rendre compte de phénomènes discursifs. Nous montrons également que le type de ressource utilisé est au moins aussi important que la façon dont on l'utilise. En effet, l'utilisation de ces représentations de mots pose certaines difficultés pour notre tâche car les observations ne sont pas des mots mais des paires de segments textuels. Il nous faut donc déterminer comment représenter un mot, un ensemble de mots et une paire d'ensembles de mots. Dans ce cadre, nous comparons différentes stratégies permettant d'obtenir un vecteur composite représentant une instance. L'utilisation de la représentation fondée sur les paires de mots des arguments présente l'avantage de nécessiter un pré-traitement minimal. Nous nous sommes cependant demandée si tous les mots étaient réellement nécessaires pour notre tâche et notamment, si l'utilisation des têtes sémantiques des arguments c'est-à-dire globalement des verbes principaux, ne pouvait pas suffire. Même si l'extraction des têtes requiert l'utilisation d'un analyseur syntaxique et la définition d'une heuristique, cette étude est intéressante dans le sens où elle peut révéler le caractère crucial de l'information lexicale. Finalement, une problématique liée à l'utilisation des ressources pré-existantes réside dans le fait qu'elles n'ont pas été conçues avec la visée de gérer des phénomènes discursifs, elles prennent en compte des dimensions cruciales pour l'analyse discursive, d'ordre syntaxique et sémantique, mais ne sont que partiellement adaptées à la tâche. Ceci correspond à un enjeu important aujourd'hui comme le montrent les travaux en classification de sentiment ayant pour visée de construire ou d'adapter une représentation de mots liée à cette tâche qui met en jeu certaines relations paradigmatiques spécifiques (MAAS et al., 2011 ; LABUTOV et LIPSON, 2013). Nous proposons dans ce cadre de construire une ressource distributionnelle dont la visée est d'associer un mot à son contexte en termes de relations de discours. Pour ce faire, nous utilisons les connecteurs comme indice d'une relation. Cette dernière méthode fait le lien entre l'approche précédente, fondée sur les données artificielles, et cette seconde approche, correspondant à une modification de la représentation.

1.3 Contributions principales

Nous concluons ce chapitre introductif en résumant les principales contributions de cette thèse pour la tâche d'identification automatique des relations discursives implicites. Elles sont présentées dans les chapitres 4 et 5. Rappelons que sur la question de savoir comment utiliser des données brutes pour notre tâche, nous proposons d'envisager deux stratégies : soit les données brutes sont annotées automatiquement en relations discursives, soit elles sont utilisées, éventuellement en se fondant sur cette annotation automatique, pour construire une représentation de mots qui reflètent une dimension paradigmatique.

Nos contributions principales sont les suivantes. D'abord, nous améliorons la compréhension des problèmes posés par la configuration reposant sur l'ajout de données artificielles (des données brutes annotées automatiquement en relation) en les considérant d'un point de vue distributionnel. Ceci nous conduit à appréhender les systèmes construits à partir de ces données comme violant une hypothèse fondamentale en apprentissage, ce qui permet à la fois de comprendre pourquoi les expériences précédentes ont mené à des résultats négatifs lorsque ces données n'étaient utilisées sans aucune forme d'adaptation (donc sans sélection ou utilisation du cadre de l'apprentissage

multi-tâche) mais aussi de proposer un cadre d'apprentissage pertinent pour cette situation, celui de l'adaptation de domaine.

Ensuite, nous proposons de nouvelles approches inspirées du cadre de l'adaptation de domaine afin de gérer les différences distributionnelles entre données naturelles et artificielles. Nous avons implémenté des stratégies mettant en œuvre une combinaison des données ou des modèles construits sur chaque ensemble de données. Nous proposons le premier système d'identification de relations implicites pour le français et présentons également des résultats pour l'anglais. Nous montrons que les méthodes d'adaptation de domaine permettent d'obtenir des améliorations importantes pour les deux langues, ce qui prouve que les données artificielles sont pertinentes à condition de les adapter aux données implicites. Nous obtenons des améliorations dépassant l'état de l'art pour l'anglais dans le cadre d'un système multiclasse. Nos expériences montrent que les données artificielles sont particulièrement utiles pour certaines relations discursives, ce que l'on peut mettre en lien avec des hypothèses linguistiques concernant la redondance des connecteurs — les relations pour lesquelles le connecteur est plus facilement redondant connaissant des améliorations plus importantes — et avec le problème en apprentissage de déséquilibre des classes — les relations les moins représentées pouvant profiter plus fortement de l'apport de nouvelles données.

Nous menons également une comparaison détaillée entre différentes modélisations des données fondées sur les mots présents dans les arguments et conduisons des expériences dans ce cadre sur l'anglais. Nous avons implémenté des outils permettant de construire les vecteurs composites représentant les instances à partir des représentations de mots utilisés en faisant varier les schémas de combinaison et les opérations de normalisation. Nous avons également construit une ressource distributionnelle à partir du corpus artificiel annoté pour l'anglais et de différentes mesures de pondération des fréquences. Nous proposons des stratégies de combinaison de représentations de mots, construites sur des données brutes, qui conservent l'information d'ordre des arguments — information cruciale pour notre tâche —, qui présentent différents degrés d'expressivité — en capturant les interactions entre les mots et en se focalisant sur certains mots des arguments —, qui reflètent des dimensions syntaxiques et/ou sémantiques des mots ou prennent en compte une dimension rhétorique, et qui correspondent à des espaces de dimensionnalité plus ou moins élevée. Cette étude étend largement les précédents travaux sur le problème d'éparpillement de la représentation pour la tâche, notamment (Rutherford et Xue, 2014), et permet d'infirmer les conclusions de ces auteurs, de rendre compte de l'importance du choix de la façon dont on combine les informations lexicales et du caractère crucial des têtes des arguments. Un résultat particulièrement important concerne le fait que les traits traditionnellement utilisés, reposant sur des ressources construites manuellement, deviennent inutiles lorsque la représentation surfacique est transformée de façon pertinente, et ce uniquement en se fondant sur des données brutes. Ceci ouvre la voie à de nouveaux travaux dont la visée sera de découvrir quelles nouvelles informations doivent être utilisées pour la tâche. Enfin, nous obtenons des résultats plus qu'encourageants en nous fondant sur une ressource distributionnelle — donc, de manière simplifiée, fondée sur des calculs fréquentiels — construite à partir des connecteurs, scores parfois supérieurs à ceux obtenus en utilisant des ressources distribuées construites à partir de systèmes complexes (de type réseau de neurones) réclamant un entraînement réputé long et computationellement coûteux. Nous confirmons donc de précédentes conclusions concernant le fait que les ressources distributionnelles, plus rapides à construire, peuvent surpasser les représentations distribuées (Lebret et Collobert, 2014), ce qui ouvre de plus la voie à de nouveaux travaux fondés sur ce type de ressource.

Au niveau des outils développés, en plus de l'implémentation des systèmes spécifiques à chacune des stratégies, nous avons notamment implémenté des outils pour lire et construire une modélisation des données du corpus français Annodis et du corpus anglais *Penn Discourse Treebank* ainsi que des systèmes d'annotation automatique en relations explicites utilisant des heuristiques pour le

français et combinant heuristiques et modèles pour l'anglais. Afin de mener une comparaison la plus complète possible avec les études existantes sur l'anglais, nous avons mis en place des systèmes de classification dans différentes configurations mettant en jeu des classifieurs binaires — méthodologie la plus répandue — ou des classifieurs multiclasses — configuration plus réaliste —, et des jeux de relations de grain plus ou moins fin. Nous prévoyons de rendre disponible le code que nous avons développé.

1.4 Organisation de la thèse

Dans le chapitre 2, après avoir présenté brièvement les différents éléments de l'analyse discursive en section 2.1, nous décrivons en section 2.2 les cadres théoriques et formalismes développés pour l'analyse discursive qui ont donné lieu ou inspiré la construction de corpus annotés. Nous présentons ensuite les principaux schémas d'annotation et les corpus associés ainsi que les points communs et divergences entre ces schémas dans la section 2.3. Nous nous intéressons ensuite en section 2.4 aux indices linguistiques des relations discursives, les connecteurs discursifs mais aussi les autres indices qui permettent d'identifier des relations implicites. Enfin, en section 2.5, nous détaillons la composition des corpus utilisés dans cette thèse, ANNODIS et le *Penn Discourse Treebank*, et présentons les inventaires de connecteurs.

Dans le chapitre 3, nous décrivons d'abord en section 3.1 les systèmes complets d'analyse automatique du discours et reportons les performances actuelles sur cette tâche ainsi que sur les sous-tâches de segmentation et d'identification des relations explicites. Nous revenons en section 3.2 sur le caractère crucial de la tâche d'identification des relations implicites et sur les difficultés de cette tâche. Nous présentons ensuite en section 3.3 les stratégies développées dans les études existantes sur cette tâche et les performances obtenues par les différents systèmes qui, comme nous le verrons, ne sont pas toujours comparables. Enfin, nous rapportons en section 3.4 des scores de référence sur le corpus anglais du *Penn Discourse Treebank* pour différents jeux de traits afin d'introduire la configuration que nous choisissons pour les expériences décrites dans les chapitres suivants.

Le chapitre 4 est consacré aux expériences fondées sur l'utilisation des données explicites dans lesquelles le connecteur est supprimé, données dites artificielles, pour améliorer l'identification des relations implicites. Après avoir rappelé le principe et les hypothèses faites par cette méthode en section 4.1, nous développons ces hypothèses en commençant par la question de la redondance du connecteur en section 4.2, redondance requise pour que les instances dans lesquelles le connecteur est supprimé puissent servir de données d'entraînement pertinentes. Nous décrivons ensuite en section 4.3 l'hypothèse d'identité distributionnelle entre les deux types de données et montrons que cette hypothèse fondamentale en apprentissage statistique n'est pas respectée dans notre configuration. Cette conclusion nous a conduite à nous intéresser au cadre de l'adaptation de domaine qui correspond à des configurations où les données d'entraînement et d'évaluation ne suivent pas la même distribution. Après avoir décrit les corpus de données artificielles construits en section 4.4, nous présentons les expériences menées en section 4.5 en décrivant les stratégies mises en place, stratégies qui s'inspirent de méthodes développées dans le cadre de l'adaptation de domaine, et les résultats obtenus sur le corpus français ANNODIS et sur le corpus anglais du *Penn Discourse Treebank*.

Finalement, dans le chapitre 5, nous présentons la seconde approche envisagée dans cette thèse, approche cherchant à gérer le problème de la représentation des données décrit dans la section 5.1. Nous présentons ensuite les différentes représentations de mots possibles en section 5.2 avant de détailler, en section 5.3, les stratégies mises en place pour construire, à partir de ces représentations, un vecteur composite pour nos instances. La section 5.4 contient la configuration des expériences

menées sur le *Penn Discourse Treebank*. La section 5.5 est consacrée à la présentation des résultats obtenus sur ce corpus. Enfin, en section 5.6, nous décrivons la construction d'une représentation distributionnelle à partir des connecteurs et les résultats obtenus en utilisant cette représentation.

2 Formalismes et corpus pour l'analyse discursive

Sommaire

L'analyse discursive a pour but de rendre compte de la cohérence d'un texte ou document et d'en fournir une interprétation. Cette cohérence repose sur des relations qui s'établissent entre les segments qui le constituent. En effet, les segments d'un texte ne s'interprètent pas isolément les uns des autres, ils sont liés les uns aux autres de manière à former un ensemble cohérent, une structure couvrant le texte. Cette structure permet de donner au document une interprétation qui n'est pas la simple somme des sens de ces parties. Il existe différents cadres d'analyse pour le niveau du document décrivant des relations pouvant s'établir à la fois au sein des phrases ou entre des phrases, adjacentes ou non. En particulier, nous nous intéressons au cadre de l'analyse discursive qui repose sur des relations dites *rhétoriques*, aussi fréquemment appelées relations *discursives* ou *de cohérence*. Même si, comme nous le verrons, il n'existe pas de consensus sur l'ensemble de ces relations, il existe des similarités entre les différents ensembles définis, avec par exemple la définition de relations de type contrastif (*Contrast*, *Concession*, *Opposition* ...) ou causal (*Explanation*, *Result* ...). Ces relations permettent de décrire le rôle joué par chaque segment dans un document. L'analyse discursive d'un document dans ce cadre correspond à la construction d'une structure formée à partir de ces relations que nous nommerons indifféremment relations discursives ou rhétoriques. La cohérence d'un document ne s'établit pas sur la base de ce seul type de lien. Elle repose sur différents indices de cohésion établissant par exemple des liens de coréférence et d'anaphore qui lient des ensembles d'entités entre elles et avec ce qui est dit sur elles. Ces liens s'inscrivent dans le cadre de la structuration d'un texte en thèmes ou topiques (le thème étant « ce dont on parle » dans un segment textuel, voir par exemple (Longo, 2013) pour

une description de cette notion) : la continuité thématique est notamment assurée par les liens de coréférence entre des entités du document. Un document peut aussi être considéré sous l'angle des éventualités (états et évènements) qui le composent, des liens qui s'établissent entre eux et avec les cadres spatiaux-temporels permettant de les situer. Les liens auxquels nous nous intéressons sont quant à eux de type sémantico-pragmatique, ils structurent un document en liant contenus sémantiques et/ou actes de parole.

Ces différents niveaux d'analyse ne sont pas indépendants, il existe des contraintes qui les lient. Les liens entre relations temporelles, anaphoriques et structure discursive ont ainsi été étudiés dans (LASCARIDES et ASHER, 1993 ; ASHER et LASCARIDES, 2003). Par exemple, si une relation discursive de type causal lie deux évènements, alors celui dénotant la cause est situé temporellement avant celui dénotant l'effet [1]. De plus, ces niveaux d'analyse peuvent se rencontrer. Ainsi, les cadres théoriques pour l'analyse discursive et les corpus annotés incluent en général des relations temporelles. Cependant, ces relations ne correspondent pas à la tâche complète de l'analyse temporelle au sens où le nombre de relations définies est bien moins important que, par exemple, dans les corpus dédiés à cette tâche : on a ainsi trois relations temporelles dans le *Penn Discourse Treebank* contre treize dans les spécifications TimeML (PUSTEJOVSKY et al., 2003b). De plus, dans ce corpus discursif, les relations ne sont annotées qu'entre éventualités, les expressions temporelles dénotant une date ou une durée ne sont pas repérées. Par contre, dans le corpus discursif français ANNODIS, certaines expressions dénotant la localisation temporelle d'évènements sont annotées, mêlant un peu plus les deux tâches. Dans le *Penn Discourse Treebank*, sont aussi annotés des liens entre entités, similaires à un lien de type coréférence mais restreints à la réalisation d'une même entité dans deux phrases adjacentes. Ces relations correspondent cependant à une annotation spécifique, elles ne sont pas à proprement parler incluses dans les relations rhétoriques.

Tous ces niveaux d'analyse reposent sur des caractéristiques du langage qui permettent aux locuteurs de spécifier les liens entre les différents segments de manière à faciliter la compréhension. Notamment, en ce qui concerne les relations discursives, un locuteur peut utiliser des conjonctions et des adverbiaux, comme *parce que* ou *mais*. Certains de ces éléments lexicaux peuvent aussi être utilisés pour expliciter des liens spatiaux-temporels ou marquer des frontières de topiques. Quand ces éléments explicitent des relations discursives, ils sont appelés *connecteurs* ou *marqueurs discursifs*. Une relation discursive lexicalisée par un connecteur est dite *explicite*, tandis qu'en l'absence de connecteur, on parlera de relation *implicite*.

Dans ce chapitre, après avoir brièvement introduit en section 2.1 les éléments de l'analyse cruciaux dans le cadre d'un système d'identification des relations discursives, nous présentons en section 2.2 les cadres théoriques et formalismes pour l'analyse discursive qui sous-tendent la grande majorité des corpus annotés au niveau discursif. En section 2.3, nous recensons les projets d'annotation existants pour différentes langues puis nous décrivons les divergences et les points communs entre les trois grands schémas d'annotation en nous appuyant sur les corpus *RST Discourse Treebank*, ANNODIS et *Penn Discourse Treebank*. Nous nous intéressons ensuite en section 2.4 aux éléments du langage permettant d'inférer une relation en séparant le cas des connecteurs discursifs, marqueurs forts des relations explicites, et les autres indices identifiés dans la littérature et notamment cruciaux pour l'identification des relations implicites. Enfin, en section 2.5, nous présentons en détail les deux corpus annotés manuellement utilisés dans cette thèse : ANNODIS et le *Penn Discourse Treebank*.

Le travail présenté dans cette thèse reste relativement neutre par rapport aux cadres théoriques et nous ne cherchons donc pas ici à recenser de manière exhaustive les difficultés théoriques liées à la description des différents éléments de l'analyse. L'analyse discursive reste aujourd'hui un champ de

1. Notons que ce n'est cependant pas toujours le cas, par exemple dans la phrase : *Laurence est heureuse parce qu'elle part à Hawaï dans un mois*.

recherche ouvert pour lequel les différentes théories concurrentes sont constamment discutées et enrichies. Une description des problématiques actuelles des cadres théoriques peut par exemple être trouvée dans (TABOADA et MANN, 2006b ; BENZ et KÜHNLEIN, 2008).

2.1 Éléments de l'analyse

Nous présentons brièvement dans cette section les différents éléments de l'analyse discursive. Tous ces éléments sont définis et discutés dans les cadres théoriques et formalismes. Les guides d'annotation des corpus précisent quant à eux les choix faits, ou les compromis nécessaires, par rapport à ces éléments.

2.1.1 Unités de discours

De manière générale, l'analyse discursive commence par l'identification des segments textuels minimaux, les blocs élémentaires de l'analyse, équivalents des mots pour l'analyse syntaxique, appelés unités discursives élémentaires (*Elementary Discourse Unit*, désormais EDU) (CARLSON et al., 2001). Ce sont des segments de texte indivisibles au sens où il n'existe pas de relation discursive liant des morceaux au sein de ces segments. Ainsi, les deux phrases en (2) provenant du document issu du corpus ANNODIS présenté en introduction, les annotateurs ont identifié quatre EDU.

(2) [Présidente de la Délégation aux droits des femmes de l'Assemblée nationale,]$_1$ [Marie-Jo Zimmermann est inquiète.]$_2$ [Si les réformes envisagées par le gouvernement pour les européennes et les régionales sont adoptées telles qu'envisagées,]$_3$ [la parité hommes-femmes en politique risque de régresser.]$_4$

La définition précise, *a priori* de ces segments est difficile. En effet, il est clair que ces unités doivent avoir une fonction par rapport au reste du texte, puisqu'elles doivent pouvoir être liées par une relation. Cependant, l'identification de ces unités minimales doit rester la plus neutre possible, pour ne pas influencer le processus d'analyse. Sinon le problème devient circulaire avec une analyse dépendante des unités et le choix des unités dépendant de l'analyse (DEGAND et SIMON, 2005 ; TABOADA et MANN, 2006b). L'identification des unités minimales devrait donc se faire avant l'analyse. Les cadres théoriques et formalismes fondent leur identification sur différents critères relativement peu clairs qui conduisent généralement à une définition au cas par cas dans les guides d'annotation. On appelle segmentation le découpage d'un document en unités élémentaires de discours.

2.1.2 Relations de discours

On identifie ensuite des relations s'établissant entre des EDU et/ou des ensembles d'unités déjà liées par des relations, appelées unités complexes de discours (*Complex Discourse Unit*, désormais CDU) [2]. L'analyse discursive est un processus récursif : partant d'un texte segmenté, des relations

2. La première mention de ce terme semble difficile à établir. Sauf erreur de notre part, il n'est pas mentionné dans (ASHER, 1993 ; ASHER et LASCARIDES, 2003), décrivant le cadre de la Segmented Discourse Representation Theory, ni dans les premières études dans le cadre de la Rhetorical Structure Theory (MANN et THOMPSON, 1988 ; MARCU, 1997b ; CARLSON et al., 2001). Dans (ASHER, 1993), il est cependant question de « complex constituents » (p. 271 par exemple), et dans (ASHER et LASCARIDES, 2003), de « complex propositions » (p. 35 par exemple) et de « complex discourse structure » (p. 170 par exemple) correspondant à un ensemble d'EDU ou de CDU liées par des relations. MARCU (1997b) mentionne également le terme de « complex text structures » qui sont construites à partir des « unités de discours » minimales (p. 260). Le terme « Complex Discourse Unit » apparaît, nous semble-t-il, pour la première fois dans (SCHAUER, 2000a ; SCHAUER, 2000b), dans le second article notamment, il est clairement défini : « en groupant récursivement des unités élémentaires, on

s'établissent entre deux EDU formant ainsi une CDU qui peut ensuite être liée à une unité, élémentaire ou complexe, de manière à former finalement une structure couvrant tous les segments. Ainsi, l'analyse de l'exemple (2), provenant du corpus ANNODIS donc s'inscrivant dans le cadre de la Segmented Discourse Theory, correspond à l'identification d'une relation *Entity Elaboration* entre les EDU 1 et 2 qui correspond au fait que l'unité 1 donne des précisions sur l'entité présente dans l'unité 2, et une relation *Conditional* entre les unités 3 et 4, la première constituant une condition pour que le contenu de 4 soit vrai. On a enfin une relation *Explanation* entre l'unité complexe formée par le triplet constitué des unités 1, 2 et de la relation *Entity Elaboration* et l'unité complexe correspondant aux unités 3, 4 et la relation *Conditional*.

La notion de structure discursive est liée au principe de cohérence. Les cadres théoriques supposent que tout discours cohérent a une structure et cherchent à rendre compte de cette cohérence en décrivant l'organisation du discours. Les relations discursives sont généralement vues comme des prédicats binaires, donc prenant deux arguments, EDU ou CDU. Selon les cadres théoriques, elles sont définies sur des critères variés qui conduisent à des jeux de relations différents parfois raffinés dans des extensions et dans les corpus discursifs. Les inventaires regroupent généralement des relations de type temporel, causal, conditionnel, additif et comparatif. Les cadres théoriques et formalismes se distinguent également au niveau des contraintes imposées sur la structure finale du document, donc la façon de lier les unités en interdisant éventuellement certaines configurations. Selon les contraintes, la structure finale obtenue est soit un arbre soit un graphe.

Enfin, notons que la segmentation comme l'identification des relations discursives peuvent éventuellement être mises en lien avec différents indices linguistiques, notamment les connecteurs ou marqueurs discursifs, qui peuvent être considérés comme des déclencheurs des relations ou des guides pour leur inférence ou des marqueurs des frontières d'EDU. Comme nous l'avons dit, la relation *Conditional* entre les unités 3 et 4 dans l'exemple (2) est ainsi signalée par le connecteur *si*. La prise en compte de ces indices et leur caractère crucial diffèrent cependant selon les cadres même si, finalement, les guides d'annotation fournissent des listes de connecteurs afin d'aider les annotateurs.

2.2 Cadres théoriques

Différents cadres théoriques et formalismes ont été développés pour décrire l'analyse discursive. Ces cadres ont pour visée de définir la nature des structures organisant un document, la manière dont elles sont construites et interprétées. Nous présentons ici les cadres ayant conduit à l'annotation de corpus au niveau discursif : la *Rhetorical Structure Theory* et la *Segmented Discourse Representation Theory*. Nous présentons également brièvement le formalisme des *Discourse Lexicalized Tree Adjoining Grammar* qui ne sous-tend pas directement de corpus mais dont certains aspects ont inspiré le développement du corpus du *Penn Discourse Treebank*.

2.2.1 *Rhetorical Structure Theory*

La *Rhetorical Structure Theory* (désormais RST) est une théorie descriptive de l'organisation des documents. Elle a été originellement proposée par MANN et THOMPSON (1988) puis enrichie notamment par les travaux de MARCU (1997b). Ce cadre théorique a été développé avec la visée

obtient des unités complexes de discours couvrant plusieurs phrases » (nous traduisons). Ces articles se placent dans le cadre de la Rhetorical Structure Theory. Le terme est ensuite utilisé dans des études se plaçant dans l'un ou l'autre de ces cadres théoriques, même si l'objet auquel il réfère n'est pas exactement le même, notamment parce que l'attachement d'une unité à une CDU dans le cadre de la Rhetorical Structure Theory correspond en fait à deux cas : l'attachement de cette unité à ce que l'on peut considérer comme la tête de l'EDU (cas des relations dites *mono-nucléaires*) ou au segment complexe (cas des relations dites *multi-nucléaires*) (ASHER et al., 2011).

de guider la génération de texte mais a été utilisé pour d'autres tâches (TABOADA et MANN, 2006a), ce point de départ n'ayant pas fortement influencé le formalisme. L'analyse est centrée sur la question des buts communicatifs, le document est vu comme un ensemble cohérent où chaque partie joue un rôle dans le but de réaliser l'intention de l'auteur, rôle décrit par des relations dites rhétoriques. Les différents segments sont liés par ces relations de manière à former une structure arborescente couvrant l'ensemble du document. Nous verrons dans la section suivante que ce cadre a en particulier conduit à l'annotation du corpus anglais *RST Discourse Treebank*.

Concernant la définition des EDU, donc la segmentation, MANN et THOMPSON (1988) indiquent que ces unités doivent avoir une fonction vis-à-vis du discours au sens de la réalisation de l'intention de l'auteur, elles sont donc liées aux buts communicatifs, définition qu'il n'est pas aisé de mettre en pratique. Les auteurs précisent donc qu'au niveau syntaxique, ce sont en général des clauses. Nous verrons que dans le corpus construit dans ce cadre, les EDU sont cependant de nature plus variée.

En RST, les segments peuvent être de deux types : *nucleus* ou *satellite*. Cette distinction correspond à la description d'un principe hiérarchique entre les segments liés par une relation, l'un, le nucleus, étant considéré comme plus important pour la compréhension du texte que l'autre, le satellite. Cette asymétrie repose sur différentes observations comme le fait que l'un des segments peut être compris indépendamment de l'autre, l'inverse n'étant pas vrai. Cette distinction entre directement dans la définition des relations en RST, elle est mise en œuvre en associant à chaque relation un schéma d'application dans lequel les arguments de la relation sont étiquetés en tant que nucleus — segment central, indispensable à la cohérence du texte —, et satellite —- correspondant à une information additionnelle. Si, selon MANN et THOMPSON (1988), la plupart des relations correspondent à un schéma nucleus-satellite ou mono-nucléaire, une relation peut également être établie entre deux segments nucleus, on parle alors de relation multi-nucléaire. Ainsi, dans l'exemple (3a), une relation *Concession* s'établit entre les segments 1 et 2, le premier étant considéré comme satellite, donc non nécessaire à la compréhension globale du texte, par rapport au second étiqueté comme nucleus. Par contre, dans l'exemple (3b), la relation *Contrast* s'établit entre deux segments de même importance, on a donc une relation multi-nucléaire. Ces exemples proviennent du site internet dédié au cadre théorique de la RST [3].

(3) a. [Tempting as it may be,]$_1$ [we shouldn't embrace every popular issue that comes along.]$_2$
b. [Animals heal]$_1$ [but trees compartmentalize.]$_2$

Cette distinction repose en partie sur les constructions syntaxiques mises en œuvre, la façon dont est présentée l'information. Ainsi, les exemples (4a) et (4b), issus du manuel d'annotation du *RST Discourse Treebank*, ont à peu près le même sens mais correspondent à des structures grammaticales différentes et leurs arguments sont étiquetés différemment : le premier, formé par une succession de deux phrases, est considéré comme mettant en jeu une relation multi-nucléaire contrastive tandis que le second, constitué d'une proposition principale et de sa subordonnée, correspond à une relation mono-nucléaire concessive. Cependant, ces deux types de relations peuvent également s'établir entre des phrases indépendantes, par exemple une relation mono-nuclaire de type concessif s'établit entre la phrase 1 et l'unité 2 constituée des deux phrases suivantes dans le texte (4c) [4]. Dans ce texte, l'unité 2 correspond au segment nucleus de la relation.

(4) a. [The earnings were fine and above expectations. . .]$_1$ [Nevertheless, Salomon's stock fell $1.125 yesterday. . ..]$_2$

3. `http://www.sfu.ca/rst/01intro/definitions.html`
4. Exemple provenant également du site dédié à la RST, cf. notre précédente.

b. [Although the earnings were fine and above expectations,]$_1$ [Salomon's stock fell $1.125 yesterday.]$_2$

c. [I personnally favor the initiative and ardently support disarmement negotiations to reduce the risk of war]$_1$ [But I don't think endorsing a specific nuclear freeze proposal is appropriate for CCC. We should limit our involvment in defense and weaponry to matters of process, such as exposing the weapon's industry's influence on the political process.]$_2$

En RST, la définition des relations s'inscrit dans un cadre descriptif visant à expliciter la présence d'un segment en lien avec les intentions du locuteur et les effets sur le lecteur. Elle est fondée sur des critères sémantiques, le contenu des arguments, et fonctionnelles, les buts communicatifs, et non syntaxiques ou lexicaux. MANN et THOMPSON (1988) précisent en effet qu'ils n'ont trouvé aucun signal non ambigu des relations, excluant donc notamment la possibilité de se fonder exclusivement sur la présence d'un connecteur discursif pour l'inférence d'une relation. La définition d'une relation correspond à un schéma d'application et à une liste de propositions ou de jugements visant à aider l'analyste à décider du déclenchement d'une relation. Les schémas indiquent donc la nucléarité des arguments mais aussi leur nombre, les relations en RST n'étant pas binaires dans l'article fondateur (MANN et THOMPSON, 1988). Le principe de binarité a cependant été ajouté dans les raffinements apportés par la suite à ce cadre théorique (MARCU, 1997b). Les propositions à vérifier correspondent à quatre dimensions : des contraintes sur le nucleus, le satellite et sur la combinaison du satellite et du nucleus, et l'effet de la relation sur les croyances du lecteur. Une définition peut ne donner lieu à aucune contrainte sur le nucleus et/ou le satellite. Nous donnons comme exemple la définition de la relation causale *Non-volitional Cause* (MANN et THOMPSON, 1988), une relation où une situation est causée par un acte non volontaire. Elle est associée au schéma et aux contraintes repris en 2.1. Dans un schéma RST, les traits horizontaux correspondent aux unités liées, le ou les traits verticaux identifient le nucleus et l'arc est étiqueté avec la relation identifiée. Une relation *Non-volitional Cause* s'établit ainsi entre les deux phrases formant le texte en (5) [5].

(5) [Since the objects in a wreck represent a single moment in time,] [they provide better chronological information than even the most carefully excavated terrestrial site.]

— Contraintes sur le nucleus : le nucleus présente une situation qui n'est pas causée volontairement.

— Contraintes sur la combinaison du nucleus et du satellite : le satellite présente une situation qui a causé, sans action volontaire, la situation présentée dans le nucleus ; sans la présentation du satellite, le lecteur ne pourrait pas connaître la cause spécifique de la situation ; la présentation du nucleus est plus centrale au but de l'auteur dans la mise en avant de la combinaison nucleus-satellite.

— Effet : le lecteur reconnaît la situation présentée dans le satellite comme une cause de la situation présentée dans le nucleus.

Figure 2.1.: Définition de la relation *Non-Volitional Cause* dans le cadre de la RST.

L'inventaire des relations en RST est ouvert, les auteurs précisant que de nouvelles relations peuvent être définies pour rendre compte de phénomènes spécifiques à un genre ou un domaine d'application. Le nombre de relations définies en RST varie donc, l'inventaire original comptant 24 relations (MANN et THOMPSON, 1988) enrichi ensuite à 30 [6]. Les 24 relations originales sont groupées en 12 classes contenant entre 1 et 5 relations. Ainsi, la relation *Non-Volitional Cause* fait

5. L'exemple provient également du site internet de la RST.
6. `http://www.sfu.ca/rst`

partie du groupe des relations causales avec son pendant résultant d'une action volontaire, *Volitional Cause*, ainsi que deux relations de direction inversée — au sens où le lieu d'expression de la cause correspond au nucleus et non au satellite [7] —, les relations *Volitional Result* et *Non-Volitional Result*, et une relation appelée *Purpose* décrivant une activité menée dans le but de réaliser, d'atteindre une certaine situation. Toutes ces relations sont mono-nucléaires. Par contre, le manuel d'annotation du *RST Discourse Treebank* définit une relation *Cause-Result* multi-nucléaire.

La construction de la structure pour un document passe par l'instanciation des schémas définis pour les relations identifiées. La structure finale est un ensemble de schémas instanciés en respectant quatre contraintes. D'abord, il doit exister un schéma instancié qui couvre l'ensemble du texte, c'est la contrainte de complétude. La contrainte de connexion quant à elle stipule qu'à l'exception du texte entier, chaque segment de texte doit être connecté à un autre segment, soit en étant une EDU soit le constituant d'un schéma instancié. Ces contraintes imposent donc une couverture totale. Ensuite, chaque schéma instancié met en lien un ensemble unique de segments de texte. Cette contrainte d'unicité n'autorise donc qu'une seule relation entre deux segments. Finalement, la contrainte d'adjacence impose l'adjacence des segments mis en lien. Cet ensemble de contraintes conduit à une structure finale arborescente. MARCU (1997b) ajoute un principe de nucléarité (ou compositionnalité) permettant d'identifier les arguments des relations dans une structure complexe : une relation dont l'un des arguments est une CDU a en fait pour argument le ou les nuclei de cette CDU. La définition d'un ensemble de saillance permet de faire remonter l'ensemble des nuclei le long d'une structure complexe. Ainsi, l'arbre RST correspondant au discours (6), issu du corpus ANNODIS, est représenté dans la figure 2.2 [8], en suivant la notation de MARCU (1997b) : les symboles N et S étiquettent les arcs pour distinguer respectivement les segments nucleus et satellite, les nœud internes correspondent aux relations s'établissant entre leurs fils, les feuilles correspondent aux segments de texte, représentés ici par une étiquette (i), et on associe à chaque nœud un ensemble de saillance $\{i(,j)\}$ correspondant aux segments nuclei de la structure complexe.

Les deux dernières contraintes, unicité et adjacence, posent problème. La contrainte d'unicité a été remise en cause dans le cadre du déclenchement simultané d'une relation sémantique et d'une relation pragmatique (MOORE et POLLACK, 1992), même si ce problème semble en fait plus général puisqu'il est possible que deux relations sémantiques s'établissent entre les mêmes arguments (BUSQUETS et al., 2001). Cette contrainte est abandonnée dans les cadres de la *Segmented Discourse Representation Theory* ou dans l'annotation du *Penn Discourse Treebank*. Quant à la contrainte d'adjacence, nous verrons que le guide d'annotation du *RST Discourse Treebank* introduit un type d'EDU dite enchâssée qui permet de la relaxer.

(6) [Les tours se sont effondrées moins de deux heures plus tard]$_1$ [entraînant l'immeuble du Marriott World Trade Center dans leur chute.]$_2$ [La tour 7 du WTC s'est effondrée dans l'après-midi]$_3$ [en raison d'incendies et des dégâts occasionnés par la chute des Twin Towers.]$_4$

2.2.2 *Segmented Discourse Representation Theory*

La *Segmented Discourse Representation Theory* (désormais SDRT) (ASHER, 1993 ; LASCARIDES et ASHER, 1993 ; ASHER et LASCARIDES, 2003) est un cadre d'analyse discursive qui étend les principes de la sémantique dynamique de la *Discourse Representation Theory* (DRT) (KAMP, 1981 ; KAMP

7. Notons que dans le manuel d'annotation du *RST Discourse Treebank*, la situation et inversée : la relation *Cause* correspond à une expression de la cause dans le nucleus, et la relation *Result* à une expression de la cause dans le satellite.

8. En suivant le manuel d'annotation du *RST Discourse Treebank*, la relation *Non-volitional Result* serait étiquetée par *Cause* et la relation *Non-volitional Cause* par *Result*.

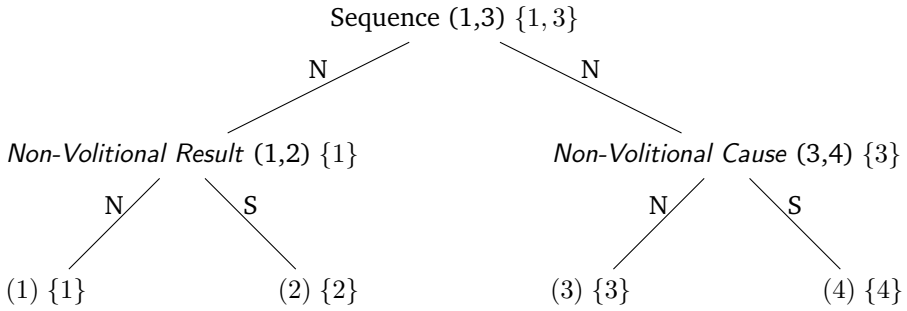


Figure 2.2.: Arbre RST pour le discours en (6).

et REYLE, 1993) et reprend la notion de structure discursive composée grâce à des relations rhétoriques comme en RST. La prise en compte des structures discursives permet, par exemple, de mieux contraindre les possibilités d'antécédents d'un pronom ou de rendre compte de l'ordre temporel d'évènements quand celui-ci ne suit pas l'ordre du texte. Les relations rhétoriques ne sont cependant pas définies en termes d'intentions comme en RST, mais de contraintes et d'effets sémantiques. La SDRT est un cadre formel qui vise l'interprétation du discours en terme logique de valeur de vérité par rapport à un modèle. Les contraintes sur la structure n'imposent pas une forme arborescente, un document peut être représenté par un graphe dirigé.

En SDRT, l'EDU est définie en termes sémantiques comme une unité contenant la description d'une éventualité, unité porteuse d'une valeur de vérité. Elle doit pouvoir être représentée en DRT sous la forme d'une DRS (*Discursive Representation Structure*). Ce sont aussi essentiellement des clauses. La DRS permet de représenter le contenu propositionnel du discours, c'est un niveau intermédiaire de représentation entre le texte et le modèle. La première étape de l'analyse, à l'interface syntaxe-sémantique, consiste donc à associer, à partir de la représentation syntaxique, une représentation sémantique, une DRS, à chaque unité élémentaire. Une DRS correspond à une paire : un ensemble d'entités — les référents du discours — est lié à des conditions sur ces référents — des formules logiques ou d'autres DRS.

Les relations discursives sont également définies sur une base logique à partir de règles déclenchant leur inférence. L'inventaire de relations est plus restreint qu'en RST, avec environ 12 relations (BUSQUETS et al., 2001). La cause d'un tel écart ne réside cependant pas tant dans des différences fondamentales entre les types de relations définies que dans une granularité plus fine du jeu de relations RST. Cette perte en granularité en SDRT est due aux exigences imposées par le cadre formel de description qui n'autorise la description d'une nouvelle relation qu'à condition de correspondre à des règles de déclenchement, des contraintes sémantiques nouvelles. Les auteurs de la SDRT ont utilisé des relations de l'inventaire de la RST mais ont défini un critère uniquement sémantique pour leur identification : des relations déclenchant une même modification de la valeur de vérité sont groupées. Ainsi, la classe de relations causales de la RST contenant 5 relations est réduite à 3 relations, *Explanation*, *Result* et *Goal*. Notamment, le fait que l'action causant une certaine situation soit volontaire ou non ne conduit pas à définir un nouveau type de relation.

Contrairement à la RST, la SDRT définit donc des règles formelles de déclenchement d'une relation fondées sur des informations sémantiques, pragmatiques et lexicales. Les règles sont associées à des effets sémantiques qui décrivent les conséquences des relations, ajoutant des informations à la représentation, et qui servent de contraintes en empêchant éventuellement l'inférence d'autres relations. Le module correspondant à ces règles modélise l'interface sémantique-pragmatique. Les relations s'établissent entre des DRS ou des ensembles de DRS liées, appelées SDRS (*Segmented DRS*). Les règles de déclenchement sont décrites dans un cadre de logique non monotone permettant

d'exprimer des règles qui vont s'appliquer par défaut. Ces règles doivent être défaisables pour tenir compte des nouvelles informations introduites lors de l'ajout d'une nouvelle EDU au discours, l'analyse étant incrémentale, les EDU sont ajoutées une à une, dans l'ordre du texte pour former la structure complète. Ainsi, alors que dans l'exemple (7a) (traduit de (LASCARIDES et ASHER, 1993)), on inférera entre les deux phrases une relation *Explanation*, la chute de Paul étant causée par le fait qu'il a été poussé, dans l'exemple (7b), on comprend que le fait d'être poussé n'est pas la cause de la chute de Paul, la chute intervient avant l'action de pousser. L'utilisation d'une logique non monotone permet de définir des règles par défaut et donc de réviser une inférence : on peut dire qu'en l'absence d'information supplémentaire, une relation de causalité sera détectée dans (7a) tandis que des informations permettent dans (7b) de réviser ce jugement et d'associer aux mêmes segments textuels la relation *Narration*.

(7) a. Paul est tombé. Marie l'a poussé.

b. Paul est tombé. Marie l'a poussé. Elle lui a ainsi évité une chute mortelle.

Les règles de déclenchement se fondent sur le contenu de la proposition en cours d'évaluation, sur le contexte, c'est-à-dire les propositions déjà rattachées et leur structure, des principes généraux (comme les principes gricéens) et des connaissances du monde. Par exemple, la présence d'un connecteur discursif peut être, contrairement à la RST, un indice suffisant de l'inférence d'une relation. Ainsi, dans le cas de deux propositions reliées par la conjonction de coordination *car*, une règle permet d'inférer la relation *Explanation*. Les règles sont exprimées à l'aide d'un prédicat ternaire $R(\alpha, \beta, \lambda)$ indiquant que α et β sont liées par la relation R dans la SDRS λ, $?(\alpha, \beta, \lambda)$ correspondant à une relation de discours encore inconnue. Dans ce cadre, la règle pour décrire l'inférence de *Explanation* pour l'exemple (8a) est donnée par la formule (8b), avec $[\text{car}](\beta)$ un prédicat indiquant la présence de *car* dans l'unité β. L'effet sémantique de cette inférence, le fait que l'évènement de pousser intervient avant la chute, est décrit par la formule (8c) avec e_α et e_β dénotant les éventualités intervenant dans les propositions, $\text{event}(e_i)$ un prédicat indiquant que e_i est un évènement et $\prec$ un opérateur de précédence temporelle entre évènements. La formule (8b) indique que la mise en lien des éventualités α et λ à l'aide du connecteur β ici identifié à *car* implique qu'une relation *Explanation* s'établit entre α et λ. La première partie de la formule (8c) précise que l'existence d'une relation *Explanation* entre deux unités entraîne la présence d'un lien de causalité entre les évènements correspondant à chacune des unités mises en jeu.

(8) a. $[\text{Max est tombé}]_\alpha$ $[\text{car Jean l'a poussé.}]_\beta$

b. $(?(\alpha, \beta, \lambda) \wedge [\text{car}](\beta)) \implies \text{Explication}(\alpha, \beta, \lambda)$

c. $\phi_{\text{Explication}(\alpha,\beta)} \implies \text{cause}(e_\beta, e_\alpha)$
$(\text{cause}(e_\alpha, e_\beta) \wedge \text{event}(e_\beta)) \implies e_\alpha \prec e_\beta$

Les connaissances du monde peuvent aussi permettre d'inférer des relations entre des évènements à travers des règles défaisables. LASCARIDES et ASHER (1993) décrivent ainsi la différence entre (7a) et (9) (également traduit de (LASCARIDES et ASHER, 1993)) par l'existence de connaissances sur le monde concernant un rapport entre *tomber* et *pousser*, tandis qu'il n'y en n'a pas sur un lien entre *se lever* et *saluer*. Comme noté par BUSQUETS et al. (2001), la définition de ces règles nécessite des analyses linguistiques fines afin d'identifier les déclencheurs des relations et de formaliser les connaissances partagées, c'est un travail long et encore en cours.

(9) Max s'est levé. Marie l'a salué.

Comme en RST, la structure discursive doit couvrir l'ensemble du document. En SDRT, la structure du discours est représentée par une SDRS similaire aux DRS définies en DRT, cadre dans lequel un discours est vrai s'il existe un modèle dans lequel le modèle représenté par sa DRS peut s'enchâsser (BUSQUETS et al., 2001). En DRT cependant, les relations rhétoriques n'interviennent pas de manière centrale. Au contraire, en SDRT, la SDRS permet de représenter l'emballage de l'information en représentant les liens entre les segments.

Une SDRS correspond formellement à un couple $< U, C >$ avec U un ensemble d'étiquettes et C un ensemble de conditions. Les étiquettes correspondent aux constituants, les unités discursives, généralement notées π_i. Une condition est une formule $\pi_i : K_{\pi_i}$, avec K_{π_i} une DRS ou une SDRS représentant l'unité d'étiquette π_i, ou une formule représentant une relation de discours R entre deux unités notée $R(\pi_i, \pi_j)$. La SDRS représentant un discours est construite incrémentalement, chaque nouvelle EDU est ajoutée, attachée à la structure et permet de mettre à jour la représentation. Cette mise à jour correspond à la représentation dynamique du sens qui se fait, selon (ASHER et LASCARIDES, 2003), par une redéfinition de la notion de sens d'une phrase : d'abord vu comme l'ensemble des modèles que la phrase satisfait, il devient une relation entre un ensemble de contextes en entrée représentant le contenu du discours avant l'unité courante, et un ensemble de contextes en sortie qui représentent le contenu du discours en incluant cette unité. Cette définition relationnelle du sens est appelée *context change potential*. Le système permet de calculer une forme logique du discours et donc de définir de façon computationnelle l'incohérence d'un discours : si aucune relation ne peut être calculée entre deux segments d'un discours, il y a incohérence.

La SDRT reprend le principe hiérarchique sur les arguments en définissant directement les relations comme *coordonnantes* ou *subordonnantes*, ce qui correspond, respectivement, à des schémas multi-nucléaires et mono-nucléaires en RST (DANLOS, 2006). Cette asymétrie permet cependant ici, en plus d'identifier des segments plus importants, de contraindre la structure en restreignant les sites d'attachement possibles pour une nouvelle EDU à travers la contrainte dite de la *frontière droite* (POLANYI, 1985). Cette contrainte stipule que seuls certains sites de la SDRS en cours de construction sont ouverts à l'attachement d'une nouvelle unité. Plus précisément, lorsqu'une relation coordonnante s'établit entre deux segments a et b, alors seul b est un site ouvert, aucun nouveau segment ne pourra être attaché à a. C'est une contrainte forte qui nécessite d'identifier précisément relations coordonnantes et subordonnantes, puisque ces types restreignent les structures possibles. Les critères d'identification de ces types peuvent notamment correspondre à la possibilité d'attachement vis-à-vis de la frontière droite ou leur compatibilité avec d'autres relations du même type (ASHER et VIEU, 2005). Cette distinction qui impose une hiérarchie sur la structure est cruciale, elle permet notamment de restreindre les référents possibles pour une anaphore : ils doivent se situer sur la frontière droite. Comme les relations SDRT sont binaires, on peut représenter les SDRS sous la forme de graphe, où les arcs, étiquetés par les relations, relient les étiquettes. Les traits verticaux représentent les relations subordonnantes et les traits horizontaux les relations coordonnantes. Une telle représentation permet de visualiser la frontière droite. On peut ainsi représenter l'exemple précédent repris en (10) dans lequel chaque EDU est représentée par une étiquette π_i par le graphe en figure 2.3 [9]. Après l'attachement de la proposition représentée par l'étiquette π_3, on ne peut plus attacher de nouvelles informations à π_1 ou π_2. Dans le graphe final, la frontière droite est constituée des étiquettes π', π_3 et π_4. Notons qu'une étude sur les données annotées dans le cadre du projet ANNODIS a montré que des annotateurs naïfs construisaient des structures respectant cette contrainte dans environ 95% des cas (AFANTENOS et ASHER, 2010).

(10) [Les tours se sont effondrées moins de deux heures plus tard]$_{\pi_1}$ [entraînant l'immeuble du Marriott World Trade Center dans leur chute.]$_{\pi_2}$ [La tour 7 du WTC s'est effondrée dans l'après-

9. La relation *Result* est généralement considérée comme coordonnante mais la rendre coordonnante ici violerait la contrainte de la frontière droite.

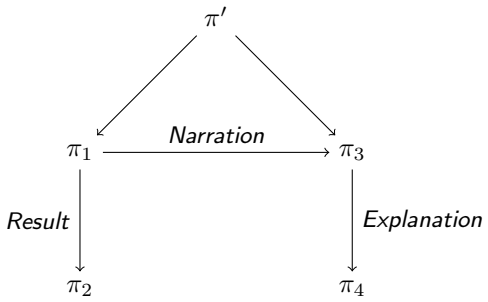


Figure 2.3.: Graphe SDRT pour le discours en (10).

midi]$_{\pi_3}$ [en raison d'incendies et des dégâts occasionnés par la chute des Twin Towers.]$_{\pi_4}$
ANNODIS, Document Attentats

2.2.3 *Discourse Lexicalized Tree Adjoining Grammar*

Le formalisme des *Discourse Lexicalized Tree Adjoining Grammar* (désormais D-LTAG) (WEBBER, 2004) étend le formalisme des *Lexicalized Tree Adjoining Grammar* (LTAG) (JOSHI, 1987 ; SCHABES, 1990) de la syntaxe aux phénomènes discursifs. Dans une grammaire LTAG, les items lexicaux, appelés *ancres*, sont représentés par des arbres dits élémentaires. Les arbres élémentaires sont de deux types : les *arbres initiaux* portent des nœuds sur lesquels peuvent se substituer d'autres arbres, les *arbres auxiliaires* portent des nœuds auxquels peuvent s'adjoindre d'autres arbres. La substitution correspond de manière générale à l'insertion d'éléments nécessaires, comme le sujet d'un verbe, et l'adjonction à une modification, par exemple par un adjectif, en introduisant une possible récursion. Le formalisme D-LTAG part de l'observation que les connecteurs discursifs peuvent fonctionner, aux niveaux syntaxique et sémantique, comme les verbes. Les connecteurs sont vus comme des prédicats prenant deux arguments, des clauses ou des ensembles de clauses liées, entre lesquels ils établissent une relation discursive. Les connecteurs et certains verbes sont donc les ancres d'arbres, initiaux ou auxiliaires, dans lesquels la racine et les feuilles autres que l'ancre représentent des unités discursives. Les clauses simples peuvent être représentées par l'arbre syntaxique TAG correspondant, ce qui permet de mener en même temps analyse syntaxique et discursive. Un arbre est défini pour chaque réalisation possible de l'ancre (nombre d'arguments, disposition ...). Les différentes opérations de combinaison d'arbres aboutissant à l'arbre dérivé sont enregistrées dans un arbre dit de dérivation. Une interface syntaxe-sémantique permet quant à elle de calculer une interprétation pour le discours par l'application de règles compositionnelles à partir de l'arbre de dérivation (FORBES-RILEY et WEBBER, 2006).

En D-LTAG, ce sont les arbres qui définissent la nature des EDU vues comme les arguments des connecteurs, ce sont donc aussi essentiellement des propositions dont la nature dépend du type de connecteur utilisé (conjonction de coordination, de subordination ou adverbial).

D-LTAG opère une distinction entre différents types de connecteurs ou autres éléments lexicaux déclenchant une relation en regard du type d'arbre qu'ils ancrent, initial ou auxiliaire, et du type de relation qu'ils déclenchent, relation de type prédicat-argument ou élaboration du discours précédent. La première catégorie correspond aux connecteurs (dont les arguments sont) structurels ancrant des arbres initiaux dont les nœuds de substitution correspondent à des clauses et déclenchant des relations prédicat-argument. Ces arbres sont ancrés par des conjonctions de subordination, des constructions parallèles (*either..or*) et certaines conjonctions de coordination. Certains verbes, comme *suppose* en forme impérative quand il introduit une condition, peuvent aussi ancrer ce type d'arbre, ainsi que certains items lexicaux appelés « subordinateurs » comme *in order for* ou *in*

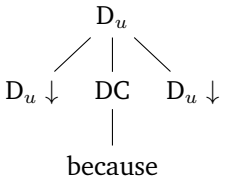
order to dont l'une des clauses est à temps non fini. Les conjonctions de coordination entrant dans ce cadre sont celles déclenchant une relation spécifique, les auteurs donnent comme exemple *so* déclenchant une relation de type *Result*. L'arbre correspondant à ce type d'ancre est représenté dans la figure 2.4a pour le connecteur *because* en position médiane avec D_u une unité discursive et DC un connecteur discursif. La sémantique liée à cet arbre est représentée sous l'arbre : l_1 est une étiquette correspondant à la représentation sémantique de la structure formée par le connecteur lié à ses arguments (i.e. unité discursive complexe). Cette représentation correspond donc, de la même manière que pour tout prédicat, à l'application de *because* à ses deux arguments représentés ici par une étiquette s_i et une adresse dans l'arbre. Les arguments de type EDU sont simplement représentés par leur formule en logique des prédicats. Par exemple, si l'argument $s1$ correspond à la clause *Jean aime Marie*, sa représentation correspond à $l_2 = \text{like}(j, m)$.

Les deux autres catégories correspondent à des items ancrant des arbres auxiliaires. Les arbres auxiliaires en TAG permettent d'introduire une récursion et de modifier des arbres élémentaires. Le premier type d'arbre auxiliaire correspond à des connecteurs structurels qui continuent une description d'une situation ou d'une entité (*and*), la continuation étant donc vue comme un ajout au discours précédent. Le connecteur vide, représentant les exemples implicites, fait partie de cette classe. Nous donnons en figure 2.4b l'arbre D-LTAG correspondant au connecteur vide ainsi que sa représentation sémantique. Le second type d'arbre auxiliaire est ancré par des adverbiaux considérés comme des connecteurs anaphoriques (comme *then*) au sens où ils prennent seulement l'un de leurs arguments structurellement tandis que l'autre est récupéré en contexte. La représentation sémantique associée à ce type de connecteur, en figure 2.4c, rend compte de cette spécificité dans le sens où seul un argument est représenté structurellement (ici s_1), l'autre, ici le second argument du prédicat *then*, sera résolu anaphoriquement en contexte. Cet argument anaphorique est représenté par une fonction d'assignation et n'est pas résolu par une compositionnalité sémantique, il doit être déterminé par un système de résolution d'anaphore.

Les relations discursives dans ce cadre sont confondues avec les connecteurs, vus comme des prédicats, ce qui ne signifie pas qu'un connecteur ne peut déclencher qu'une seule relation. Des informations supplémentaires de type sémantique, lexicale ou syntaxique doivent être prises en compte pour permettre d'inférer la relation déclenchée par le connecteur ou la relation implicite liant deux propositions. En D-LTAG, la structure se construit comme dans les grammaires TAG par substitution ou adjonction d'unités discursives, ou plutôt des arbres leur correspondant, dans les arbres représentant les connecteurs. Des règles de compositionnalité sémantique permettent de construire une interprétation pour le discours. Des contraintes peuvent être utilisées pour restreindre les arbres possibles, notamment en utilisant un critère existant en TAG (*flexible direction of composition* (JOSHI et al., 2003)) qui bloque certaines possibilités de composition. On obtient, à la fin de l'analyse d'un discours, un arbre représentant le niveau syntaxique de l'analyse, une formule logique correspondant à sa représentation sémantique et une structure de dérivation, généralement de type arborescente mais qui peut éventuellement être un graphe (FORBES-RILEY et WEBBER, 2006).

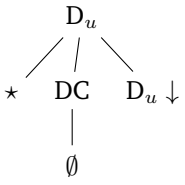
Un autre formalisme nommé *Discourse Synchronous* TAG (D-STAG) a été développé par DANLOS (2009). Il étend au discours les TAG synchrones permettant de lier deux niveaux d'analyse dont les opérations se font simultanément. Dans D-STAG, la représentation en syntaxe du discours est menée en même temps que le calcul de son interprétation. Ce formalisme diffère sur plusieurs points de D-LTAG. Notamment, D-STAG reprend le principe hiérarchique sur les relations de la SDRT ce qui permet de conserver les contraintes imposées sur la structure définies dans ce cadre. Notons que ce formalisme fait actuellement l'objet d'une traduction vers le formalisme des Grammaires Catégorielles Abstraites (ACG) (DANLOS et al., 2015).

(a) Arbre élémentaire pour *because*



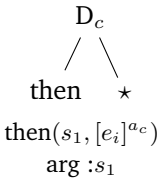
l_1 : because(s_1, s_2)
arg : $< s_1, (1) >, < s_2, (3) >$

(b) Arbre auxiliaire pour le connecteur vide.



$l_1 : \emptyset(s_1, s_2)$
arg : $s_1, < s_2, (3) >$

(c) Arbre auxiliaire pour *then*.



then$(s_1, [e_i]^{a_c})$
arg :s_1

Figure 2.4.: Types d'arbres élémentaires en D-LTAG.

2.3 Corpus discursifs

Les cadres théoriques et formalismes présentés précédemment ont conduit, plus ou moins directement, à l'annotation de corpus au niveau discursif. Le cadre de la RST a notamment conduit à l'annotation du *RST Discourse Treebank* pour la langue anglaise et celui de la SDRT au corpus ANNODIS pour la langue française. Le formalisme D-LTAG ne sous-tend pas directement l'annotation d'un corpus, mais le principe de se fonder sur les connecteurs pour construire une représentation discursive d'un document a inspiré le développement du *Penn Discourse Treebank*. Les différents corpus présentent des différences qui reflètent les divergences entre les cadres théoriques mais qui sont également dues à des visées différentes ou à des choix nécessaires dans la définition d'un schéma d'annotation. En particulier, les corpus construits dans les cadres de la RST ou de la SDRT sont assez similaires au niveau du processus d'annotation et du type d'annotation produit tandis que le *Penn Discourse Treebank* diffère sur ces points.

Nous décrivons d'abord brièvement les projets d'annotation menés pour différentes langues s'inscrivant dans le cadre de la RST, dans celui de la SDRT ou suivant le modèle du *Penn Discourse Treebank*. Nous présentons également les quelques corpus annotés en-dehors de ces schémas. Nous revenons ensuite sur les points communs et différences entre les schémas d'annotation en focalisant notre description sur les corpus *RST Discourse Treebank*, ANNODIS et *Penn Discourse Treebank* puisque ce sont les corpus les plus utilisés dans les systèmes automatiques et, pour les deux derniers, ceux que nous utilisons dans nos expériences. Ces deux derniers corpus seront par ailleurs présentés de manière plus complète à la fin de ce chapitre.

2.3.1 Corpus construits dans le cadre de la RST

Le cadre théorique de la RST a donné lieu à l'annotation du premier corpus discursif (MARCU et al., 1999) visant à évaluer la possibilité de construire des analyses RST manuellement et automatiquement. Ce corpus comporte 90 documents issus de différents corpus (MUC7, Brown-Learned et Wall Street Journal). Il n'a pas été réutilisé par la suite pour des systèmes automatiques mais a permis de montrer la possibilité d'annoter des documents dans ce cadre, malgré la difficulté de la tâche, et d'élaborer un premier guide d'annotation. Il a mené à l'annotation du *RST Discourse Treebank* (désormais RST DT) (CARLSON et al., 2001) contenant 385 documents du *Penn Treebank* (MARCUS et al., 1993) déjà annotés manuellement en syntaxe. C'est aujourd'hui le corpus de référence pour la construction de systèmes complets d'analyse discursive. Ce projet a aussi mené à l'annotation de corpus dans ce cadre pour d'autres langues grâce notamment au développement d'un outil d'annotation, le RSTTool [10]. Il existe ainsi des corpus pour l'espagnol, le *Spanish RST Discourse Treebank* (CUNHA et al., 2011), l'allemand pour la sous-partie MAZ du *Postdam Commentary Cor-*

10. http ://www.sfu.ca/rst/06tools/

pus (STEDE, 2004), le portugais, avec le corpus *Rhetalho* [11] (PARDO et SENO, 2005) ou le basque, pour le *RST Basque Treebank* (IRUSKIETA et al., 2013). Un corpus a aussi été développé pour un domaine spécifique, le *SFU Review Corpus* en version anglaise et espagnole [12]. Enfin, notons qu'un projet multilingue [13] regroupe 15 documents parallèles annotés en anglais, espagnol et basque correspondant à des résumés d'articles scientifiques. Ce corpus a permis une comparaison des structures dans différentes langues et une étude des procédés de traduction à ce niveau (IRUSKIETA et al., 2015).

2.3.2 Corpus construits dans le cadre de la SDRT

Pour la langue anglaise, le corpus DiSCoR (REESE et al., 2007) a été annoté suivant le cadre de la SDRT. Ce corpus avait pour visée d'étudier les interactions entre structures rhétoriques et phénomènes de coréférence. C'est dans le cadre de la SDRT qu'a été annoté le premier, et à ce jour le seul, corpus discursif pour la langue française ANNODIS (AFANTENOS et al., 2012a). L'annotation de ce corpus a mené au développement d'un outil d'annotation, Glozz [14] (PÉRY-WOODLEY et al., 2009 ; WIDLÖCHER et MATHET, 2009), assez générique pour permettre l'annotation des données à différents niveaux — ce corpus contient en effet l'annotation, en plus de celle de grain fin des relations rhétoriques, de structures d'organisation discursive plus globale (notamment la structure énumérative). Cet outil permet également la visualisation de la structure discursive complète, de type graphe, du document. Un corpus a aussi été annoté dans ce cadre pour la langue arabe, le *Arabic Discourse Treebank* (KESKES et al., 2014). Tous ces corpus ont la particularité d'être de taille relativement restreinte (86 document pour ANNODIS, 90 pour le *Arabic Discourse Treebank* et 60 pour DISCoR). Un corpus supplémentaire a été annoté pour la langue anglaise et le domaine particulier des discussions instantanées mettant en jeu plus de deux participants dans des conversations stratégiques autour d'un jeu de plateau (AFANTENOS et al., 2012b). Ce cadre d'annotation semble constituer une difficulté plus grande pour les annotateurs (GASTEL et al., 2011), même si le manque de corpus annotés dans ce cadre pourrait simplement résider dans un manque de moyen, l'annotation discursive étant particulièrement longue et difficile.

La SDRT proposant un formalisme logique intéressant pour une interprétation des textes et de certains phénomènes, on peut se demander s'il est possible de transformer des annotations existantes vers ce cadre. Pour les jeux de relations discursives, ROZE (2013) propose des correspondances entre notamment les jeux de relations définies en RST, en SDRT et dans le corpus du *Penn Discourse Treebank*. BENAMARA et TABOADA (2015) s'intéressent aux jeux de relations définies dans les cadres de la RST et de la SDRT et aux relations annotées dans les corpus RST DT, ANNODIS, DISCoR et dans le *Arabic Discourse Treebank* et proposent une taxonomie unifiée sur les cadres et les guides d'annotation qui ouvre la voie à des travaux intéressants fondés sur différents corpus. La question de correspondances entre structures est plus complexe, d'abord à cause des différences en termes de segmentation mais aussi parce que ces différents cadres n'ont pas les mêmes capacités représentationnelles (DANLOS, 2006). Notons cependant que des travaux sont menés pour regrouper sous un même langage différents types de structure discursive (VENANT et al., 2013).

2.3.3 Modèle du *Penn Discourse Treebank*

Le corpus du *Penn Discourse Treebank* (désormais PDTB) (MILTSAKAKI et al., 2004), comme nous l'avons dit, ne s'inscrit pas dans un cadre théorique spécifique. Il se distingue notamment par une analyse plus locale des phénomènes discursifs, l'annotation étant fondée sur des éléments lexicaux

11. `http://www.icmc.usp.br/~taspardo/projects.htm`
12. `http://www.sfu.ca/~mtaboada/research/SFU_Review_Corpus.html`
13. `http://ixa2.si.ehu.es/rst/`
14. Disponible sur la page `http://glozz.free.fr/`.

particuliers, les connecteurs discursifs. Nous revenons plus en détail sur ces principes dans la section suivante. Ce corpus a été enrichi de nouvelles annotations dans sa seconde version (PRASAD et al., 2008a). Il correspond aujourd'hui à 2 259 documents, il est donc de taille largement supérieure aux corpus s'inscrivant dans les cadres RST ou SDRT. Comme le RST DT, il est annoté sur des données du PTB. Ce corpus est le seul, pour la langue anglaise, à distinguer dans l'annotation relations implicites et explicites, il est donc le plus utilisé pour la tâche d'identification des relations implicites. Il a notamment permis de montrer les différences en termes de performance pour la tâche d'identification des relations selon la présence ou l'absence d'un connecteur discursif. Il a conduit au développement d'un outil d'annotation, Annotator [15], au moins réutilisé pour le projet sur le français. Ce modèle d'annotation a donné lieu à de nombreux projets dans des langues très diverses. Ainsi, des corpus de type PDTB ont été développés ou sont en cours de développement pour le français, *French Discourse Treebank* (désormais FDTB) (DANLOS et al., 2012 ; STEINLIN et al., 2015), le tchèque (POLÁKOVÁ et al., 2013), l'hindi (PRASAD et al., 2008b), le turc (ZEYREK et WEBBER, 2008 ; ZEYREK et al., 2009), l'arabe (AL-SAIF et MARKERT, 2010) et le chinois (LI et al., 2014b). Un corpus a aussi été développé pour la langue anglaise et le domaine bio-médical (PRASAD et al., 2011). Une comparaison entre ces différents corpus, à l'exception du corpus français, peut être trouvée dans (PRASAD et al., 2014). Notons aussi que le corpus allemand, le *Postdam Commentary Corpus*, dans lequel les relations sont annotées suivant le cadre RST, contient une annotation des connecteurs et de leurs arguments similaire à celle proposée dans le PDTB (STEDE, 2004 ; STEDE et NEUMANN, 2014). Le succès de ce modèle d'annotation tient en partie au fait qu'il est plus facile à appréhender pour les annotateurs notamment parce que l'annotation se fonde sur des éléments lexicaux et ne requiert pas l'identification d'une structure globale pour un document.

2.3.4 Corpus constitués dans un autre cadre

Le corpus du *GraphBank* (WOLF et GIBSON, 2005) est, à notre connaissance, le seul corpus discursif pour la langue anglaise construit en-dehors des trois cadres présentés précédemment. Les 11 relations rhétoriques utilisées proviennent essentiellement de l'inventaire de HOBBS (1985). L'une des visées de ce corpus était de montrer que la structure d'arbre, qui est imposée en RST, n'est pas suffisante pour représenter toutes les structures discursives possibles. La présence de dépendances croisées et de nœuds ayant plusieurs parents impose une structure de graphe moins contrainte. Le corpus contient 135 documents, des articles du *Wall Street Journal* et du *AP Newswire*. Pour l'allemand, GASTEL et al. (2011) ont annoté au niveau discursif 31 articles journalistiques issus du corpus *TüBa-D/Z* (TELLJOHANN et al., 2009), en s'inspirant des cadres RST, SDRT et du corpus du PDTB.

2.3.5 Points communs et divergences entre les schémas d'annotation

Il existe donc, globalement, trois grandes familles de schémas d'annotation de corpus discursifs : celui fondé sur le cadre de la RST, celui suivant le cadre de la SDRT et le modèle du PDTB. Nous nous intéressons dans cette section aux points communs et différences entre ces trois schémas en fondant notre description sur trois corpus les instanciant : le RST DT, ANNODIS et le PDTB.

2.3.5.1 Processus d'annotation et information encodée

De manière générale, une première divergence entre ces schémas est observable au niveau du processus d'annotation, différence importante car elle se reflète dans les systèmes complets d'analyse discursive construits à partir de ces corpus comme nous le verrons dans le chapitre suivant. Le processus d'annotation est commun aux corpus suivant le cadre RST ou SDRT, il correspond à la

15. https://www.seas.upenn.edu/~pdtb/tools.shtml#annotator

segmentation intégrale du texte en EDU puis à l'identification des relations entre les segments en respectant éventuellement certaines contraintes sur la structure. L'annotation est incrémentale : les annotateurs relient les EDU pour former des CDU qui sont liées les unes aux autres de manière à couvrir l'ensemble du document. Notons que l'outil d'annotation du RST permet de voir la structure se compléter au fur et à mesure de l'annotation et de vérifier ainsi le respect de contraintes définies dans ce cadre. Pour ANNODIS, les annotateurs n'étaient pas forcés de respecter la contrainte de la frontière droite, le développement du corpus permettant plutôt d'en vérifier la validité. L'outil d'annotation Glozz permet cependant également de visualiser le graphe discursif obtenu. L'annotation d'un corpus du type du PDTB suit un processus différent, l'annotation commençant par l'identification des connecteurs discursifs, de leurs arguments et des relations qu'ils déclenchent. Les annotateurs identifient ensuite des relations entre les segments non liés par un connecteur. Ces segments sont considérés comme les arguments de relations non explicites. Cette différence implique notamment que la notion d'EDU n'existe pas vraiment dans le PDTB où elle est remplacée par la notion d'argument. De plus, l'annotation est locale, les annotateurs n'ont pas à s'assurer d'une mise en relation de toutes les parties d'un texte entre elles ce qui, cependant, facilite l'annotation et explique en partie la popularité de ce schéma d'annotation et l'amélioration des scores d'accord.

Une autre différence importante réside dans les informations encodées. Notamment dans le PDTB sont annotés différents types de relations distinguant globalement les relations selon qu'elles sont explicitées ou non par un connecteur, distinction que l'on ne retrouve pas dans le RST DT ou dans ANNODIS. Dans le PDTB, la présence d'un connecteur conduit à l'annotation d'une relation de type *explicite*. Dans la seconde phase d'annotation, les annotateurs étiquettent donc des relations non explicites. Là-encore, les connecteurs jouent un rôle, ils permettent de distinguer plusieurs cas conduisant à quatre autres types de relation :

— si les annotateurs peuvent introduire un connecteur entre les segments, la relation est dite *implicite*,

— si l'insertion d'un connecteur paraît introduire une forme de redondance par rapport à une autre expression, la relation est dite *lexicalisation alternative*, elle est en fait lexicalisée mais par une forme qui ne correspond pas à un connecteur,

— si aucune des deux conditions précédentes n'est vérifiée mais que les annotateurs identifient la mise en lien de deux entités, la relation est dite *relation d'entité*,

— et, finalement, une étiquette marquant l'*absence de relation* est utilisée quand aucun autre type de relation n'est identifié.

Ces distinctions entre différents types de relation sont l'une des richesses de l'annotation du PDTB. Pour l'instant, l'annotation de ces liens non explicites est limitée aux phrases adjacentes au sein d'un même paragraphe ainsi qu'aux propositions séparées par deux points ou un point virgule. L'annotation n'est donc pas complète, il manque notamment les relations implicites intra-phrastiques et inter-paragraphes. Ce sont ces hypothèses simplificatrices qui ont conduit à l'introduction d'un lien de type absence de relation car, bien sûr, cette étiquette ne signifie pas que les segments ne sont reliés à aucune autre unité dans le texte, ce qui correspondrait à une incohérence, mais simplement que la relation qui les lie n'est pas explicite et n'est pas locale.

Nous reprenons ci-après des exemples des différents types de relation provenant du manuel d'annotation du PDTB (PRASAD et al., 2007) : dans l'exemple (11a) une relation explicite est établie entre les deux propositions et le connecteur *because* est repéré comme marqueur de cette relation ; dans l'exemple (11b), une relation implicite est identifiée entre les deux phrases et l'annotateur propose le connecteur *so* comme marqueur de la relation, cette information étant conservée dans l'annotation ; dans l'exemple (11c), l'annotateur a considéré que l'insertion d'un connecteur introduirait une redondance, une relation de type lexicalisation alternative (AltLex)

est donc annotée et l'expression introduisant la redondance, ici *after that*, est annotée ; dans l'exemple (11d), une relation de type entité (EntRel) est identifiée mettant en lien les expressions « *Hale Milgrim* » et « *Mr. Milgrim* », aucun élément lexical n'est ajouté à l'annotation de la relation ; enfin, dans l'exemple (11e), l'annotateur n'a identifié aucune relation entre les phrases qui sont donc liées par l'étiquette dénotant l'absence de relation (NoRel).

(11) a. [The federal government suspended sales of U.S. savings bonds]
because $_{\text{Explicit}}$ [Congress hasn't lifted the ceiling on government debt.]

b. [The projects already under construction will increase Las Vegas's supply of hotel rooms by 11,795, or nearly 20%, to 75,500.] *so* $_{\text{Implicit}}$ [By a rule of thumb of 1.5 new jobs for each new hotel room, Clark County will have nearly 18,000 new jobs.]

c. And she further stunned her listeners by revealing her secret garden design method : [Commissioning a friend to spend "five or six thousand dollars ... on books that I ultimately cut up."]$_1$ *After that* $_{\text{AltLex}}$, [the layout had been easy.]$_2$

d. [Hale Milgrim, 41 years old, senior vice president, marketing at Elecktra Entertainment Inc., was named president of Capitol Records Inc., a unit of this entertainment concern.] $\emptyset_{\text{EntRel}}$ [Mr. Milgrim succeeds David Berman, who resigned last month.]

e. Jacobs Engineering Group Inc. 's Jacobs International unit was selected to design and build a microcomputer-systems manufacturing plant in County Kildare, Ireland, for Intel Corp. [Jacobs is an international engineering and construction concern.] $\emptyset_{\text{NoRel}}$ [Total capital investment at the site could be as much as $400 million, according to Intel.]

Notons enfin que le traitement de l'attribution — les cas où le contenu d'une proposition est attribué à une source à travers l'utilisation d'un verbe de communication ou d'attitude propositionnelle (« dire », « penser » . . .) — est différent selon les corpus, ce qui conduit à la présence d'informations différentes. Dans le RST DT et dans ANNODIS, une relation spécifique dénommée *Attribution*, incluse dans le jeu de relations discursives, permet d'encoder le fait qu'un segment est attribué à un agent : la relation lie le contenu du message rapporté à la source et au verbe introduisant ce discours. Un exemple provenant du corpus ANNODIS est donné en (12a) et un exemple du corpus RST DT en (12b), ces exemples provenant des manuels d'annotation respectifs de ces corpus (MULLER et al., 2012b ; CARLSON et MARCU, 2001). Dans ces deux exemples, une relation *Attribution* est annotée entre les segments.

(12) a. [La direction générale de Citroën a informé ses employés] [que les nouveaux contrats de travail prendront effet lundi prochain.]

b. [The legendary GM chairman declared] [that his company would make "a car for every purse and purpose."]

Dans le PDTB, l'attribution correspond à un niveau d'annotation différent de celui des relations discursives. Les informations concernant l'attribution sont marquées pour les relations de type explicite, implicite ou lexicalisation alternative en identifiant le segment attributif, comportant généralement un verbe de discours ou une expression dénotant un discours rapporté comme « *according to* », et en ajoutant des traits selon quatre dimensions : la source de l'attribution (entre l'auteur du texte, un agent introduit dans le texte ou un individu non spécifique), le type d'attribution (dénotant un degré de factualité entre assertion, croyance, fait et intention/attitude), la polarité (entre négative et positive, une négation pouvant inverser la polarité de la relation ou du contenu d'un argument [16]), et le caractère déterminé ou non de l'attribution (le caractère indéterminé

16. Notons que, dans le même ordre d'idée, le corpus RST DT définit une relation *Attribution-n* marquant une attribution négative.

	REL	Arg1	Arg2
[SOURCE]	Arb	Inh	Inh
[TYPE]	PAtt	Null	Null
[POLARITÉ]	Null	Null	Null
[DÉTERMINÉ]	Indet	Null	Null

Figure 2.5.: Structure de traits pour l'attribution dans l'exemple (13a) issu du PDTB.

correspond au fait que l'attribution peut être annulée dans certains contextes spécifiques comme un contexte conditionnel). Pour chacune de ces dimensions, l'annotation précise si l'attribution porte uniquement sur l'un des arguments ou sur la relation entière. Ainsi, l'exemple (13a) provenant du manuel d'annotation du PDTB contient le segment attributif « *to think* », il est associé à la structure de traits en figure 2.5. Dans cet exemple, la relation est attribuée à un individu non spécifique, qui n'intervient pas dans le texte et considéré comme arbitraire (« Arb »), les valeurs « Inh » pour les arguments indiquant que ceux-ci héritent de cette valeur pour ce trait. Cette attribution est de type croyance avec l'utilisation d'un verbe d'attitude propositionnelle (« PAtt »), la polarité est positive et l'attribution peut éventuellement être annulée dans certains contextes, elle est ici uniquement conjecturée (« Indet »). Le segment attributif « *to think* » est annoté comme information supplémentaire en-dehors des arguments ce qui est le cas de tout segment attributif contenant un verbe à moins que le segment intervienne directement dans l'inférence de la relation. Par contre les segments attributifs non propositionnels sont inclus dans l'empan des arguments comme c'est le cas pour « *according to* » dans l'exemple (13b) provenant du manuel d'annotation du PDTB, le segment étant cependant repéré comme segment attributif dans l'annotation. Le PDTB propose donc une annotation bien plus riche et plus fine de ces phénomènes que les autres corpus. Notons que jusqu'à présent, l'attribution n'est pas prise en compte directement dans les systèmes automatiques bien qu'elle soit très certainement d'une importance capitale (PRASAD et al., 2014 ; HUNTER et DANLOS, 2014).

(13) a. It is silly libel on our teachers **to think** [they would educate our children better] if only [they got a few thousand dollars a year more.]

b. [No foreign companies bid on the Hiroshima project, **according to** the bureau.] *But* [the Japanese practice of deep discounting often is cited by Americans as a classic barrier to entry in Japan's market.]

2.3.5.2 Segmentation

Le problème de la détermination des unités à lier par des relations discursives, considéré généralement comme relativement simple et quasiment résolu dans les systèmes automatiques, est en fait tout aussi complexe que crucial. La nature des segments discursifs nous intéresse particulièrement dans le cadre d'un système d'identification des relations discursives puisqu'ils correspondent aux entrées du système statistique. La difficulté, au niveau de la segmentation, est de déterminer la frontière entre les informations syntaxiques, sémantiques et discursives. En général, la phrase est considérée comme une EDU, elle peut même être le seul type d'EDU reconnu, et certains systèmes d'analyse automatique du discours se sont limités aux relations inter-phrastiques. On peut en effet penser que, puisque le discours est censé gérer des phénomènes qui dépassent le cadre de la phrase, contrairement à la syntaxe et à certains aspects de la sémantique, on pourrait se limiter à ce type d'unité. Mais on risque de perdre ainsi en granularité de l'analyse. Le problème est de conserver une cohérence sur ce que l'on doit considérer comme appartenant au domaine de l'analyse discursive. Ainsi les exemples (14a) à (14d), proposés par CARLSON et al. (2003), ont à peu près le même sens mais ce sens est exprimé en deux phrases distinctes, en deux clauses ou en une seule clause.

(14) a. [Xerox Corp.'s third-quarter net income grew 6.2% on 7.3% higher revenue.] [This earned mixed reviews from Wall Street analysts.]

b. [Xerox Corp's third-quarter net income grew 6.2% on 7.3% higher revenue,] [which earned mixed reviews from Wall Street analysts.]

c. [Xerox Corp's third-quarter net income grew 6.2% on 7.3% higher revenue,] [earning mixed reviews from Wall Street analysts.]

d. [The 6.2% growth of Xerox Corp.'s third-quarter net income on 7.3% higher revenue earned mixed reviews from Wall Street analysts.]

Si l'on considère que l'on a clairement une relation rhétorique, de type *Consequence*, entre les deux phrases dans l'exemple (14a), on voudrait idéalement conserver cette analyse quelle que soit la forme que prend l'expression de cette relation en syntaxe. La segmentation proposée de ces exemples dans le manuel d'annotation du RST DT montre une tendance à favoriser la proposition, le dernier exemple ne contenant qu'une seule proposition, une seule EDU et donc pas de relation rhétorique. Les différents cadres théoriques donnent de ces segments des définitions relativement différentes mais surtout peu précises, la délimitation des EDU demeure un problème généralement traité au cas par cas dans les manuels d'annotation.

Si l'on se réfère aux définitions des cadres et des manuels, quelques consensus émergent cependant. Concernant la nature des EDU, les phrases et les propositions complètes, contenant la description d'une éventualité réalisée avec tous ces arguments, sont toujours des EDU. Les manuels définissent ensuite pour chaque autre cas les possibilités de segmentation. En général, toutes les propositions subordonnées, à l'exception des relatives restrictives, sont segmentées, de même que les compléments de verbes de communication ou d'attitude propositionnelle [17]. Il faut noter cependant que même le consensus concernant la phrase n'en est pas tout à fait un. Pour un corpus s'inscrivant dans le cadre de la RST ou de la SDRT, c'est bien le cas. Cependant, dans le PDTB, la segmentation correspond essentiellement à l'identification des arguments des connecteurs. Or, l'argument d'un connecteur peut être un ensemble de phrases. Si ces phrases peuvent être liées par la suite par une relation rhétorique ce n'est toutefois pas toujours le cas. Elles peuvent en effet également être liées par une étiquette dénotant l'absence de relation ou par un lien indiquant uniquement que les phrases réfèrent à une même entité. De plus, une autre spécificité dans l'annotation de ce corpus est le *principe de minimalité* [18] (PRASAD et al., 2006). Selon ce principe, seuls les morceaux de texte minimalement nécessaires à l'inférence de la relation sont inclus dans l'argument, le reste étant éventuellement annoté comme matériel supplémentaire. Ainsi, dans l'exemple (15), le segment « *Workers described "clouds of blue dust"* » est considéré comme matériel supplémentaire de l'argument 1. Ce second point exclut généralement des morceaux de phrase d'un argument, et donc, à moins de considérer comme EDU l'union du matériel nécessaire et des segments supplémentaires, la phrase n'est pas, dans ce corpus, considérée comme une EDU. Plus généralement, ce concept induit une couverture non totale : contrairement au RST DT ou à ANNODIS, tous les segments textuels ne sont pas reliés les uns aux autres, ils ne sont pas tous forcément inclus dans l'argument d'une relation. Notons qu'une analyse partielle est déjà intéressante et pourrait se révéler utile pour différentes applications de TAL. De plus cette caractéristique se justifie sur certains points comme la non inclusion de certains segments attributifs et des connecteurs.

(15) ($_{Sup1}$ Workers described "clouds of blue dust") [that hung over parts of the factory]$_1$ *even though* [exhaust fans ventilated the area.]$_2$

17. L'attribution correspond à une annotation particulière dans le PDTB que nous avons présentée dans la section 2.3.5.1

18. Cette notion apparaît dans le premier manuel d'annotation du corpus mais pas dans les premiers articles (MILTSAKAKI et al., 2004 ; PRASAD et al., 2004) qui font cependant déjà référence à l'annotation de matériel supplémentaire pour les arguments.

D'autres consensus reposent sur l'organisation des EDU entre elles : les EDU ne peuvent pas se chevaucher mais on peut avoir des EDU enchâssées. Ce dernier point a donné lieu à une forme d'accommodation pour un corpus s'inscrivant dans le cadre de la RST pour lequel une contrainte impose l'adjacence des EDU. Cependant, certaines clauses considérées comme des EDU peuvent être exprimées à différentes places dans une phrase, éventuellement au milieu d'une autre clause. Pour pallier ce problème, les auteurs du RST DT ajoutent une relation *Same-Unit* permettant de lier les morceaux discontinus de l'EDU principale et des relations généralement de type *Elaboration-embedded* pour lier les EDU enchâssées à l'EDU principale. Entre les EDU enchâssées au sein d'une même EDU principale, des relations classiques peuvent être annotées. Pour ce corpus, on a donc en fait deux types d'EDU clairement séparés dans le manuel de segmentation. Par ailleurs, le fait d'être réalisée au sein d'une autre EDU n'est pas une condition nécessaire pour être annotée comme EDU enchâssée dans le RST DT, c'est aussi le cas des segments textuels qui modifient une partie seulement d'une EDU, comme une relative ou une expression parenthétique. Nous donnons ci-dessous des exemples d'EDU enchâssées dans les corpus ANNODIS, exemple (16a), et dans le RST DT, exemples (16b) et (16c), en reprenant les conventions adoptées dans les guides d'annotation respectifs : l'EDU 1 est enchâssée dans l'EDU 2 dans l'exemple (16a), ce qui est marqué par le fait qu'un crochet est ouvert pour débuter l'EDU 2 juste avant le mot « *jusqu'à* » et qu'il n'est pas refermé avant l'ouverture d'un nouveau crochet marquant le début de l'EDU 1 ; dans les exemples (16b) et (16c) du RST DT, les EDU enchâssées sont signalées par une taille de police plus petite dans le manuel et, dans le cas où une EDU est rendue discontinue par la présence de l'EDU enchâssée comme dans (16c), l'EDU discontinue est en gras. Dans ce dernier exemple, une relation *Same-Unit* permet d'indiquer les segments discontinus d'une même EDU.

(16) a. Bientôt 10% des nœuds de BITNET dans le monde jouaient à MAD, [jusqu'à ce que les administrateurs de BITNET, [effrayés par le succès du jeu,]$_1$ demandent à l'École des Mines de le faire arrêter]$_2$

b. [Mr. Volk, 55 years old, succeeds Duncan Dwight,]$_1$ [who retired in September.]$_2$

c. [**But maintaining the key components of his strategy**] [– a stable exchange rate and high levels of imports –] **will consume enormous amounts of foreign exchange.**]

Enfin, la possibilité de s'appuyer ou non sur les éléments syntaxiques ou lexicaux diffère théoriquement, que ce soit pour identifier les frontières d'EDU ou la relation déclenchée. MANN et THOMPSON (1988) déclarent ainsi que le marquage lexical ou syntaxique des propositions n'est pas nécessaire à la description des structures discursives. Cependant, comme le montrent les exemples précédents (14a) à (14d), c'est bien l'organisation en syntaxe de l'information qui décide de la segmentation. Finalement, ces indices ont bien été utilisés pour décider de la segmentation dans le RST DT, CARLSON et al. (2003) autorisant même l'annotation de syntagmes à condition qu'ils débutent par un connecteur discursif comme l'unité 3 dans l'exemple (17) issu du manuel.

(17) [But some big brokerage firms said]$_1$ [they don't expect major problems]$_2$ [*as a result of* margin calls.]$_3$

Pour la construction du PDTB, les annotateurs n'avaient pas accès à l'analyse syntaxique mais seulement au texte brut afin de ne pas calquer la segmentation sur la syntaxe. Cependant, puisqu'ils cherchaient les arguments des connecteurs dont au moins un dépend syntaxiquement du connecteur, il est clair que ces informations ont été prises en compte. De plus, après l'annotation des arguments des connecteurs, les annotateurs ont eu pour tâche d'identifier une relation non explicite entre les phrases adjacentes non liées par un connecteur. Les arguments de ces relations correspondent donc à une unité syntaxique, même si le principe de minimalité a pu mener à une segmentation

dépassant ou au contraire diminuant cette unité. Finalement, que ce soit pour le RST DT, ANNODIS ou, bien sûr, le PDTB, des listes de connecteurs discursifs ont été fournies aux annotateurs.

Dans l'exemple (18), nous présentons un exemple de segmentation dans le corpus ANNODIS pour deux phrases issues d'un même document. On peut y voir que sont effectivement segmentées les propositions principales 1 et 3, ainsi que les subordonnées participiale 2, comportant un participe présent, et causale 4, ne contenant pas de verbe mais introduit par une forme de connecteur ou proche d'un connecteur [19]. Dans le RST DT, la segmentation serait la même car la proposition participiale est un modifieur de l'ensemble de la clause 1 (par contre, les propositions participiales ne modifiant qu'une partie de la clause sont segmentées en EDU enchâssée). Le cas de *en raison de* entre quant à lui dans la catégorie des unités discursives marquées explicitement, on a donc bien également une segmentation de l'unité 4 [20]. Dans le PDTB, la segmentation étant fondée sur les connecteurs en ce qui concerne la séparation d'une phrase en plusieurs EDU, les EDU 1 et 2 seraient fusionnées. Les segments 3 et 4, même en supposant que *en raison de* soit considéré comme un connecteur, seraient probablement également fusionnés. Sur ce genre de cas, le manuel n'accepte comme EDU que les nominalisations qui permettent clairement une interprétation existentielle, ce qui ne nous semble pas un critère totalement clair [21]. Un point important concerne l'insertion ou non du connecteur dans l'argument. Comme montré dans l'exemple (18), le connecteur est inclus dans ANNODIS mais aussi dans le RST DT. Dans le PDTB au contraire, il n'est pas inclus, comme on peut le voir dans l'exemple (11a). Il correspond à une information supplémentaire dans l'annotation ce qui rend compte du fait qu'un connecteur ne fait pas partie du contenu propositionnel, il est la marque du lien entre les segments.

(18) [Les tours se sont effondrées moins de deux heures plus tard]$_1$ [entraînant l'immeuble du Marriott World Trade Center dans leur chute.]$_2$ [La tour 7 du WTC s'est effondrée dans l'après-midi]$_3$ [en raison d'incendies et des dégâts occasionnés par la chute des Twin Towers.]$_4$
ANNODIS, Document Attentats

2.3.5.3 Relations discursives

Concernant les relations discursives, nous avons vu que les inventaires de la RST et de la SDRT étaient construits sur des critères différents et étaient en conséquence de taille différente. De plus, le jeu de relations défini dans un cadre théorique est souvent modifié ou raffiné lors de l'annotation d'un corpus. Le corpus RST DT comporte ainsi 78 relations, dont 53 sont mononucléaires et 25 multi-nucléaires. Ces relations peuvent être regroupées en 16 classes sur la base de similarités rhétoriques. L'inventaire d'ANNODIS contient 18 relations, regroupées en relations coordonnantes ou subordonnantes, en 8 ensembles de relations de sémantique similaire, que nous détaillerons en section 2.5.1.1, ou en 4 classes correspondant globalement à celles du PDTB (MULLER et al., 2012a) c'est-à-dire : *Temporal*, *Contingency*, *Comparison* et *Expansion*. Parmi ces 18 relations, 9 sont subordonnantes et 7 sont coordonnantes. Les deux relations restantes, *Frame* et *Temploc* (localisation temporelle), n'existent pas dans le cadre théorique originel. Nous donnons des exemples de ces relations dans la section 2.5.1.1. La relation *Frame* semble pouvoir être coordonnante et subordonnante si l'on se réfère à la structure discursive rapportée dans (PERY-WOODLEY et al., 2011). Le PDTB a la particularité de proposer une hiérarchie de relations à trois niveaux comportant 4 classes divisées en 16 types eux-mêmes éventuellement divisés en 23 sous-types. Les quatre classes au sommet de la hiérarchie correspondent à des groupes de relations que

19. L'existence de connecteur introduisant des syntagmes nominaux est sujet à caution. Ainsi, dans le FDTB, cette forme n'est pas considérée comme un connecteur
20. Exemple du manuel d'annotation : [*Any liquid water formed in this way would evaporate almost instantly*] [*because of the low athmospheric pressure.*]
21. Exemple du PDTB : *(they) are hoping* [*for major new liberalizations*] *if* [*he is returned firmly to power.*], intéprété comme *that there will be major new liberalizations*.

l'on retrouve dans les autres cadres (ROZE, 2013). Les types et sous-types correspondent également en général à des relations que l'on retrouve dans d'autres cadres avec par exemple dans la classe *Contingency* un type *Cause* divisé en deux sous-types *Reason* et *Result*, correspondant globalement aux relations *Cause* ou *Explanation* et *Result* définies dans ANNODIS ou dans le RST DT.

La hiérarchie du PDTB a notamment l'avantage d'autoriser des sous-spécifications : en cas de doute sur l'annotation d'une relation, l'annotateur peut remonter dans la hiérarchie et annoter seulement avec un type ou une classe. Ce principe permet de simplifier l'annotation mais pose quelques questions dans le cadre d'un système de classification puisque le jeu de relations est, quelque part, constitué de l'ensemble de tous les sous-types, types et classes. Nous verrons dans le chapitre suivant que les systèmes automatiques se placent à un niveau spécifique de la hiérarchie en ignorant éventuellement les instances sous-spécifiées.

L'annotation d'un corpus dans une nouvelle langue a pu conduire à une modification de l'inventaire des relations. Nous n'avons pas connaissance d'études proposant des relations propres à une langue, par contre, des relations peuvent être ajoutées en lien avec des principes théoriques ou pour augmenter le grain de l'analyse. Dans le *French Discourse Treebank* par exemple, la hiérarchie du PDTB est modifiée notamment afin d'introduire une nouvelle classe de relations non véridicales correspondant aux cas où le contenu d'au moins un des arguments est hypothétique (DANLOS et al., 2012).

En plus des quatre grands groupes communs aux différents inventaires, on trouve dans tous les jeux de relations une distinction entre des relations appelées relations sémantiques ou thématiques en RST, liant le contenu sémantique des segments, et des relations appelées pragmatiques dans le PDTB, présentationnelles en RST ou méta-relations en SDRT, qui lient un acte de parole et le contenu sémantique d'un argument. Dans ANNODIS ces deux types de relation sont clairement séparés avec par exemple la possibilité d'annoter une relation de type *Explanation*, liant le contenu sémantique des arguments, et une relation de type *Explanation** pour un lien avec un acte de parole. Dans le PDTB, ces types sont aussi séparés avec la définition de types sémantiques, comme *Cause*, et de types pragmatiques, comme *Pragmatic Cause*.

Nous avons déjà évoqué une différence en termes de granularité entre les jeux de relation définis en RST et en SDRT, une différence qui se reflète dans les corpus. De plus, il existe des relations qui n'ont été introduites que dans certains projets d'annotation, sans ce que cela corresponde, du moins uniquement, à une question de granularité. Ainsi, le jeu de relation défini pour le corpus ANNODIS contient une relation appelée *Frame* qui n'a pas, à notre connaissance, d'équivalent dans d'autres corpus discursifs. Cette relation met en lien un adverbial détaché en tête de proposition, introduisant un cadre localisant notamment temporellement ou spatialement une situation, et le segment sur lequel porte ce cadre. L'introduction de cette relation correspond également à une spécificité en termes de segmentation dans ce corpus : tous les adverbiaux, quand ils sont détachés en tête de proposition, sont segmentés. Un exemple de ce type est donné en (19a) dans lequel « *le lendemain* » est segmenté. Sont aussi segmentés les syntagmes comportant un nom d'évènement ou d'état introduit par une préposition ou locution prépositionnelle spatiale, temporelle, causale, concessive ou contrastive. Concernant les expressions temporelles, le manuel du RST DT indique au contraire qu'elles ne sont segmentées que si elles apparaissent dans des constructions clausales, aucun syntagme temporel n'est segmenté en EDU même s'il contient une forme sémantiquement de type évènementiel. On peut voir ainsi ces principes de segmentation différents dans les exemples (19a) provenant du manuel d'ANNODIS et (19b) provenant du manuel du RST DT. L'annotation du RST DT ne contient donc pas de relations temporelles s'établissant entre des segments non propositionnels.

(19) a. [Le lendemain,] [M. Pitoun est retrouvé sain et sauf,] [**après quinze jours de séquestration.**]

b. [**Just a week after the company's dismissal of several hundred employees,** further layoffs were announced.]

L'ajout de cette relation d'encadrement participe d'une divergence plus générale au niveau du traitement des relations temporelles. On peut identifier en général trois types de liens temporels dans les corpus discursifs : (1) les situations décrites se déroulent simultanément (ce qui englobe généralement les cas de recouvrement partiel), (2) les situations décrites se produisent dans leur ordre d'apparition dans le texte ou (3) dans l'ordre inverse. Le PDTB est le plus simple sur ce point : la classe *Temporal* contient le type *Synchronous* correspondant au cas (1), et le type *Asynchronous* regroupant les deux autres cas et se divisant en deux sous-types, *Succession* pour (2) et *Precedence* pour (3). Les deux autres corpus contiennent en fait plus de relations reflétant un lien temporel entre les arguments. Dans ANNODIS, la relation *Background* correspond au cas (1), *Narration* au cas (2) et *Flashback* au cas (3). Cependant, sont aussi définies les relations *Frame*, pour les adverbiaux éventuellement temporels, et *Temploc* (localisation temporelle) qui correspond à l'annotation de subordonnées temporelles ou d'adverbiaux comportant un nom d'éventualité comme le segment 2 dans l'exemple (20a) issu du manuel d'annotation d'ANNODIS (MULLER et al., 2012b). Elle se distingue donc de *Frame* en forçant la présence de l'expression d'une éventualité dans le segment exprimant une localisation et parce qu'elle est limitée aux phénomènes temporels. La localisation temporelle peut marquer indifféremment une concomitance entre des éventualités, comme dans l'exemple (20b), ou une asynchronie comme c'est le cas dans l'exemple (20a). Il nous semble donc qu'ici, le choix d'une relation dépend en partie de la réalisation en syntaxe des arguments : contrairement à (20b), l'exemple (20c) serait plutôt annoté par la relation *Narration*.

(20) a. [Paul est arrivé]$_1$ [après le départ de Marie.]$_2$

b. Dimanche en milieu d'après-midi, [quelques enfants s'adonnaient à des glissades] [alors que l'eau, cachée, recouvrait encore tout le secteur.]

c. [Paul est arrivé.]$_1$ [Ensuite, Marie est partie.]$_2$

Dans le RST DT, la relation *Sequence* correspond au cas (2) et la relation *Inverted Sequence* au cas (3) lorsque les situations sont des évènements et que la relation est multi-nucléaire. Lorsque la relation temporelle est mono-nucléaire, et notamment, selon le manuel, quand les arguments sont réalisés en syntaxe par une proposition subordonnée et sa principale, on utilisera plutôt les relations *Temporal-After* pour (2) et *Temporal-Before* pour (3). Ainsi, l'exemple (21a) est annoté avec la relation *Inverted Sequence* tandis que l'exemple (21b) correspond à la relation *Temporal-Before*, ces exemples étant issus du manuel d'annotation du RST DT. Ceci rejoint nos observations précédentes sur le fait que la nucléarité paraît souvent confondue avec la réalisation en syntaxe des arguments, plutôt que d'être liée au type de relation comme cela semblait être originellement prévu.

(21) a. [Three new issues begin trading on the New York Stock Exchange today,] [and one began trading on the Nasdaq/National Market System last week.]

b. [We want to make sure they know what they want] [before they come back.]

De plus, le traitement du cas (1) dans le RST DT nous paraît peu clair. En effet, il existe une relation *Temporal-Same-Time*, à la fois mono- et multi-nucléaire, qui doit prendre en compte les cas de synchronie. Cependant, il existe également deux relations, *Background* et *Circumstance*, définies par : « la situation présentée dans le satellite fournit le contexte dans lequel la situation

présentée dans le nucléus doit être interprétée » (CARLSON et MARCU, 2001) (nous traduisons). Nous donnons ci-dessous un exemple de *Background* en (22a) et de *Circumstance* en (22b), les exemples provenant du guide d'annotation du corpus. Le guide précise que *Circumstance* est « plus forte » que *Background* car, pour la seconde, « l'information ou le contexte n'est pas toujours spécifié clairement ou fortement délimité ». De plus, si les situations sont concomitantes, on préfèrera la relation *Circumstance*, tandis que si les situations se déroulent à des moments différents, on annotera plutôt *Background*[22]. La distinction entre ces deux relations nous semble assez floue, mais on retiendra que *Circumstance* a une définition similaire de celle de la relation *Background* dans ANNODIS.

(22) a. [The Voting Rights Act of 1965 was enacted to keep the promise of the fifteenth Amendment and enable Southern blacks to go to the polls (...).] [Twenty-five years later, the Voting Rights Act has been transformed by the courts (...) into a program of racial gerrymandering (...)]

b. [The project appeared to be on the rocks earlier this month] [when the other major partner in the project (...) was to have a 40% stake in Luzon Petro- chemical.]

On observe donc une représentation des phénomènes temporels relativement différente selon les corpus, avec des raffinements correspondant notamment au fait que les situations mises en lien sont ou non des évènements, ou de la réalisation en syntaxe de l'expression temporelle. On note que BENAMARA et TABOADA (2015), qui proposent un jeu de relations unifié à partir, en particulier, des trois corpus que nous étudions ici, aboutissent à une classe temporelle contenant trois relations : *Sequence*, *Inverted Sequence* et *Synchronous*. Les relations *Frame*, *Background* et *Circumstance* sont fusionnées en deux relations, *Circumstance* disparaissant, et groupées dans une sous-classe *Framing* d'un groupe plus large appelé *Thematic*. Leur séparation des relations temporelles rend compte du fait que le cadre peut être également spatial ou refléter un domaine (par exemple « en médecine » est un cadre « de domaine » segmenté dans ANNODIS (MULLER et al., 2012b)). Il reste cependant à savoir si, dans cette classification, les phénomènes temporels liés, par exemple, à une relation causale, seront doublement annotés : dans l'exemple canonique repris en (23), doit-on annoter *Explanation* et *Inverted Sequence* ? La définition assez lâche des relations temporelles du PDTB encourage une telle double annotation, mais étant donné le faible taux d'annotation multiple, on peut se demander si ce phénomène a été pris en compte de manière exhaustive. Plus généralement, étant donné qu'il existe toujours une relation temporelle entre des éventualités liées, il pourrait être intéressant de séparer les niveaux rhétoriques et temporels et de proposer une surcouche d'annotation temporelle riche, du type de celle proposée dans un corpus comme le TimeBank (PUSTEJOVSKY et al., 2003a), au-dessus des annotations des relations discursives.

(23) Paul est tombé parce que Marie l'a poussé.

Les inventaires de relations sont donc assez différents entre les corpus même s'ils présentent des points communs et que, comme nous l'avons évoqué, des correspondances peuvent s'établir entre les relations définies. Un autre point de divergence entre les corpus concerne la possibilité d'annoter plusieurs relations entre les mêmes arguments. Le cadre de la RST impose une contrainte d'unicité, une seule relation peut donc être annotée entre deux arguments dans le RST DT. Ceci a conduit cependant à la mise en place, dans le manuel d'annotation du RST DT, d'un protocole qui définit un ordre d'annotation des relations afin d'exprimer des préférences et d'assurer une consistance dans le choix des annotateurs en cas de doute. ANNODIS et le PDTB autorisent au contraire des

22. Ces distinctions temporelles n'apparaissent cependant pas dans (MANN et THOMPSON, 1988).

annotations multiples. Il est clair que plusieurs relations peuvent s'établir en même temps entre deux évènements, et notamment quand on inclut des relations temporelles dans le jeu de relations rhétoriques. C'est par exemple le cas dans l'exemple (24) repris de (ASHER et LASCARIDES, 2003) (nous traduisons) où une relation *Contrast* s'établit entre les deux arguments, marquée par le connecteur *but*. Une relation de type *Narration* est cependant également présente de manière implicite entre les deux unités : cette relation permet d'assurer l'identification de l'ordre temporel correct entre les évènements d'achat et de location.

(24) [Jean a acheté un appartement] [mais il l'a mis en location.]

Forcer les annotateurs à faire un choix alors que plusieurs relations sont clairement inférables paraît donc contre-intuitif. Il est cependant possible qu'il existe des règles permettant de retrouver les relations non annotées à partir de celles annotées, par exemple en définissant une algèbre des relations de discours du type de celle proposée par ROZE (2013). Une autre étude qu'il serait intéressant de mener consisterait à comparer les relations annotées dans le RST DT et dans le PDTB pour leur partie commune. Par ailleurs, notons que si le PDTB autorise des annotations multiples, il n'est pas clair que toutes les relations possibles aient été annotées étant donné le faible taux, environ 2% des données (PRASAD et al., 2014), de relations multiples.

Dans l'exemple du corpus ANNODIS repris en (25), une relation de type *Result* est identifiée entre les segments 1 et 2, l'effondrement des tours étant la cause de la chute de l'immeuble, une relation *Narration* lie l'EDU 3 au discours précédent, une CDU correspondant à la phrase constituée des EDU 1 et 2, et une relation *Explanation*, relation inverse de *Result*, lie les EDU 3 et 4. Dans le RST DT, on aurait une relation *Cause* entre les segments 1 et 2 car ici le résultat est dans le segment satellite, la relation *Result* étant réservée aux cas où le segment le plus important contient l'effet, ce qui est le cas pour la relation entre les segments 3 et 4[23]. L'enchaînement temporel entre la première et la seconde phrase serait représenté par la relation multi-nucléaire *Sequence*. Pour le PDTB, si l'on admet qu'il n'y a pas de connecteur ici, on aura uniquement une relation temporelle de type *Succession* entre les deux phrases. On peut se demander quelle relation serait annotée entre 1 et 2 puisque l'annotation d'un exemple non explicite passe, dans le cadre actuel c'est-à-dire inter-phrastique, par l'insertion d'un connecteur entre les arguments, ce qui semble difficile ici.

(25) [Les tours se sont effondrées moins de deux heures plus tard]$_1$ [entraînant l'immeuble du Marriott World Trade Center dans leur chute.]$_2$ [La tour 7 du WTC s'est effondrée dans l'après-midi]$_3$ [en raison d'incendies et des dégâts occasionnés par la chute des Twin Towers.]$_4$
ANNODIS, Document Attentats

2.4 Indices linguistiques des relations discursives

Nous avons vu que l'identification des relations, et d'ailleurs des segments élémentaires, se fait selon différents principes et indices selon les cadres théoriques. Dans cette section, nous nous intéressons aux indices linguistiques qui permettent de signaler la présence d'une relation discursive et de l'identifier. Ces indices sont cruciaux dans le cadre de la construction d'un système automatique d'identification des relations discursives puisqu'ils correspondent aux informations que l'on voudra modéliser. Parmi ces indices, certaines formes, regroupées sous le nom de marqueur ou connecteur discursif, ont une importance particulière. Ils sont ainsi au centre d'un formalisme comme D-LTAG et considérés, en tout cas pour une partie d'entre eux, comme des indices forts d'une relation en

23. Rappelons que ces relations sont inversées par rapport à leur définition dans le cadre de la RST.

SDRT. De plus, nous l'avons dit, tous les manuels d'annotation fournissent pour chaque relation une liste de marqueurs discursifs potentiels afin de guider les annotateurs.

Le PDTB est le premier corpus à avoir fourni une annotation identifiant les connecteurs discursifs et séparant, de ce fait, les cas de relations explicites, pour lesquels un connecteur est identifié dans le texte, et les cas de relations implicites, pour lesquels D-LTAG construirait un connecteur vide. Ceci a permis de construire des systèmes automatiques dédiés à chacun de ces types de relation et de montrer des différences de performance très importantes. Nous présentons ces scores en détail dans le chapitre suivant. Ces scores reflètent la difficulté de la tâche d'identification des relations implicites liée à l'absence de l'indice fort constitué par le connecteur et à la complexité des autres indices potentiels. Dans le reste de cette section, nous revenons sur les connecteurs discursifs et présentons ensuite les autres indices linguistiques identifiés dans la littérature.

2.4.1 Les connecteurs discursifs

Dans cette section, nous nous intéressons à la question de la définition des connecteurs et reportons au chapitre suivant la question de leur caractère nécessaire, problème qui est au centre de la méthode fondée sur l'utilisation de données explicites pour l'identification des relations implicites. Rappelons que nous nous limitons au cadre des monologues écrits, les dialogues et le langage parlé introduisant des problématiques différentes.

Les connecteurs discursifs peuvent être considérés de manière générale comme des éléments particuliers de la langue dont la fonction est de lier deux segments textuels par une relation sémantique et de structurer le texte. L'existence de telles marques est une notion généralement acceptée. Cependant, ils ont donné lieu à des définitions très diverses (Taboada, 2006) ce qui résulte en des inventaires de connecteurs différents. Nous nous restreignons ici aux études qui lient les connecteurs et les relations discursives telles qu'elles sont définies dans les cadres théoriques présentés dans les sections précédentes. Cela signifie que seront considérés comme connecteurs des éléments liant des unités acceptées comme unités discursives. Ainsi, les connecteurs peuvent lier des propositions à l'intérieur des phrases mais aussi des phrases, ils participent donc à la segmentation en séparant les unités discursives intra-phrastiques et à la structuration globale du texte. De plus, les connecteurs discursifs doivent pouvoir déclencher une relation rhétorique. Dans ce cadre, les connecteurs, comme les relations discursives, sont généralement vus comme des prédicats à deux arguments, ils correspondent à une lexicalisation des relations (Jayez et Rossari, 1998 ; Webber, 2004). On considère qu'ils forment une catégorie plus ou moins fermée et peuvent être de catégories morpho-syntaxiques différentes : conjonctions de coordination (*et, ou, mais*) et de subordination (*parce que, comme, bien que*), adverbiaux (*ainsi, ensuite, donc*), prépositions (*afin de, avant de*), éventuellement certains verbes et syntagmes nominaux (*la preuve*). Cette diversité en termes de catégorie morpho-syntaxique induit des différences en termes de fonctionnement, et ce d'abord en syntaxe. Ainsi, si l'on appelle clause hôte le segment dans lequel le connecteur apparaît, et clause conviée le segment constituant son second argument (Danlos, 2009), chaque catégorie de connecteur est liée à certaines contraintes sur leur disposition (Roze et al., 2012). La clause conviée d'une conjonction de coordination ou d'un adverbial se situe toujours à gauche de la clause hôte (clause hôte postposée) tandis que, pour les conjonctions de subordination, la clause hôte peut être à gauche, à droite (préposée) ou au sein (médiane) de la clause conviée comme on peut le voir dans les exemples (26a-26c) repris de (Roze, 2009). De plus les adverbes peuvent se trouver en position initiale, médiane ou finale au sein de leur clause hôte.

(26) a. Pierre est rentré tôt *parce qu'* il est fatigué.

b. *Parce qu'* il était fatigué, Pierre est rentré tôt.

c. Pierre, *parce qu'* il est fatigué, est rentré tôt.

La spécificité des adverbiaux a aussi pu être analysée en d'autres termes. Ainsi, nous avons vu que dans D-LTAG étaient distingués des connecteurs structurels, les conjonctions, et des connecteurs anaphoriques, les adverbiaux. Cette typologie se fonde par exemple sur le fait que seuls les connecteurs adverbiaux admettent des dépendances croisées, une caractéristique partagée avec les anaphores (WEBBER et al., 2003). Cette distinction conduit à des différences sur la façon dont les connecteurs sont liés à leurs arguments : les connecteurs structurels récupèrent leurs deux arguments de manière structurelle tandis que les adverbiaux ne récupèrent que l'un de leurs arguments de manière structurelle, l'autre étant fourni anaphoriquement en contexte. Ceci permet, selon (WEBBER et al., 2003) de simplifier les représentations et de maintenir une compositionnalité sémantique pour le discours. Ce lien entre connecteur et anaphore a cependant été remis en cause dans (DANLOS, 2009 ; ROZE, 2009), les anaphores correspondant selon ces études à des phénomènes différents. Les connecteurs adverbiaux ont cependant bien des particularités, comme de pouvoir être précédés par un autre connecteur (*mais, néanmoins*) et d'autoriser des connexions entre des segments distants.

JAYEZ et ROSSARI (1998) font quant à eux une distinction entre des connecteurs *temporels* correspondant à des relations entre éventualités comme la succession, la simultanéité ou la précédence, des connecteurs *thématiques* qui associent ou mettent en contraste des informations partageant un type informationnel et connectent des segments de structure similaire, et enfin les connecteurs *inférentiels* qui informent sur les inférences à prendre en compte et la façon de les combiner pour obtenir une interprétation. Cette distinction permet de rendre compte du fait que les connecteurs sont hétérogènes au niveau du type d'information qu'ils manipulent. Elle est enrichie par une autre distinction reposant sur l'observation que, tandis que certains connecteurs relient directement le contenu propositionnel de leurs arguments (*du coup, de ce fait*), certains peuvent porter sur les actes de parole (*donc, alors*), déclenchant, en SDRT, une relation de type pragmatique (JAYEZ et ROSSARI, 1998 ; MOESCHLER, 2002). La distinction de connecteurs temporels apparaît dans d'autres études. Notamment, BRAS (2008) distingue plusieurs types de marqueurs temporels, permettant de séparer les simples localisateurs d'éventualités de marqueurs qui ont une influence sur la structure du discours et qui seront donc inclus dans les connecteurs discursifs (ROZE, 2013). La propriété d'organisation structurelle des connecteurs est aussi mise en avant dans des organisations s'intéressant à leur fonction argumentative et en fait des indices « permettant d'accéder à la représentation du discours » (MOESCHLER, 2002). Finalement, concernant leurs effets, les connecteurs permettent de guider l'interprétation. MOESCHLER (2002) montre ainsi qu'un connecteur peut avoir différents effets selon le contexte, confirmant une inférence, révélant l'incohérence d'un discours ou déclenchant une relation que l'on n'aurait pu inférer en son absence. Rappelons qu'en SDRT, certains marqueurs discursifs entrent dans la définition d'axiome qualifiée de règle dure, ce qui signifie que leur présence déclenche l'inférence de la relation considérée plus ou moins indépendamment du contenu des propositions reliées. C'est le cas en français de *puis* pour la relation *Narration* ou de *car* pour la relation *Explanation*. Ces effets ont permis de définir des critères d'identification des connecteurs pour constituer des lexiques comme le lexique des connecteurs du français, LexConn (ROZE et al., 2012).

La présence d'une forme correspondant à un connecteur discursif n'est pas suffisante pour identifier une relation. D'une part, les relations ne sont pas toujours lexicalisées par un connecteur et, d'autre part, ces formes sont ambiguës. On distingue en général deux types d'ambiguïté. D'abord une forme recensée comme connecteur n'est pas toujours utilisée en emploi discursif. On parlera d'*ambiguïté en emploi*. Ce problème regroupe des distinctions relativement simples, comme le fait que *et* est en emploi discursif quand il lie des propositions mais pas quand il coordonne des groupes nominaux, et

des questions plus complexes, comme la limite entre argument sous-catégorisé et argument discursif étudiée par exemple dans le cas de la préposition *pour* en français dans (COLINET et al., 2014) et plus généralement dans la construction de la première version du FDTB, le FDTB1 (STEINLIN et al., 2015). Le FDTB1 comporte à ce jour l'annotation de 10 429 connecteurs correspondant à 353 formes, un nombre largement supérieur à l'inventaire du PDTB (100 formes) ce qui a permis de mettre au jour de nombreuses difficultés pour cette tâche. L'ambiguïté en relation concerne des connecteurs pouvant déclencher des relations différentes, cette ambiguïté dépendant de l'inventaire de relations considéré et donc du cadre théorique et/ou du corpus.

Un lexique de connecteurs est constitué d'une liste d'expressions pouvant apparaître en emploi discursif et peut aussi éventuellement préciser la ou les relations déclenchées par chaque connecteur. Des inventaires de connecteurs ont été construits pour différentes langues dont l'anglais (KNOTT, 1997 ; FORBES-RILEY et WEBBER, 2006), le français (ROZE et al., 2012), l'allemand (STEDE et UMBACH, 1998) et existent pour toutes les langues pour lesquelles un projet d'annotation type PDTB a été développé. Leur taille varie, KNOTT (1997) construit un ensemble de 350 formes, LexConn en contient 328, la construction du lexique allemand de STEDE et UMBACH (1998) s'est originellement concentrée sur 170 connecteurs fréquents tandis que le PDTB s'appuie sur seulement 100 formes. Notons que VERSLEY (2010) a proposé d'induire automatiquement un tel inventaire à partir de corpus parallèles anglais-allemand, la partie anglaise étant annotée automatiquement au niveau des connecteurs grâce au PDTB. Ce type de travaux ouvre la voie à la construction de systèmes d'identification automatique des relations explicites pour les langues non dotées d'inventaires de connecteurs. Ceci participe du caractère crucial de la compréhension du lien entre relations explicites et implicites, puisqu'un système capable d'identifier les relations implicites à partir des explicites permettrait de généraliser l'analyse discursive aux langues peu dotées. Dans cette thèse, les connecteurs discursifs pris en compte proviennent de deux sources : la base des connecteurs discursifs du français (LexConn) construite par ROZE et al. (2012) et les connecteurs utilisés dans l'annotation du PDTB (PRASAD et al., 2008a). Dans LexConn comme dans l'inventaire du PDTB, les connecteurs peuvent appartenir à quatre catégories morpho-syntaxiques : les conjonctions de coordination, les conjonctions de subordination, les prépositions introduisant un verbe à l'infinitif et des adverbiaux regroupant certains syntagmes prépositionnels. Si ROZE et al. (2012) définissent des critères et des tests pour identifier les connecteurs, les choix effectués pour construire l'inventaire de connecteurs du PDTB sont moins clairs. Nous décrivons la façon dont ces inventaires ont été construits dans la section 2.5.2.

2.4.2 Autres indices

Les connecteurs ne sont pas les seuls indices de déclenchement d'une relation, ce qui permet bien sûr d'identifier des relations non lexicalisées par un tel marqueur, les relations implicites. Par ailleurs, notons dès à présent que certaines relations sont plus rarement marquées par un connecteur que d'autres. Ainsi, TABOADA et MANN (2006b) recensent différentes études de corpus dans le cadre RST qui ont déterminé que les relations *Enablement*, *Evaluation*, *Elaboration* et *Solutionhood* n'étaient jamais marquées, les relations *Background* et *Summary* étant quant à elles rarement marquées. Il doit donc exister d'autres indices dans la langue permettant de signaler une relation. Plusieurs études se sont intéressées à ce problème, notamment TABOADA et DAS (2013) ont conduit une étude sur 40 articles annotés dans le RST DT et ont identifié des informations de type morphologique, syntaxique, sémantique, lexical et graphique.

Les indices lexicaux recouvrent des phénomènes assez différents. Ainsi, ils peuvent correspondre à des formes qui ressemblent aux connecteurs discursifs mais qui ne sont pas considérées comme tels pour différentes raisons, comme les lexicalisations alternatives du PDTB qui regroupent des

formes contenant un élément anaphorique ou des expressions non figées et compositionnelles comme « *that compares with*, *one reason is*,... ». Certains verbes peuvent également être des indices forts, comme le verbe « *concéder* » pour *Concession* et « causer » pour *Cause*. La présence de nombre, montant en monnaie ou pourcentage par exemple, ou de termes comparatifs (« *plus fort*, *meilleur*,... ») peut également signaler des relations comparatives, et les dates, jours de la semaine ou mois, des relations temporelles. Comme nous l'avons dit, la SDRT notamment fonde l'inférence des relations sur différentes informations. Ces informations ne permettent pas en général de décider de l'inférence d'une relation mais guident l'interprétation vers une telle inférence. Ainsi, la présence de certains verbes, leur temps, leur mode, leur contenu sémantique en lien avec des connaissances du monde, peut amener à inférer une relation. Nous avons vu dans la section 2.2.2 que les connaissances partagées permettaient d'inférer une relation causale entre les deux phrases dans l'exemple (27a). Ici, c'est donc la présence de la paire de verbes *tomber, pousser* qui permet d'inférer la relation. Bien sûr, cette inférence se fonde sur des connaissances partagées sur un lien entre ces deux verbes, connaissances difficiles à modéliser. Nous verrons au cours de cette thèse les solutions proposées dans le cadre de la construction de systèmes automatiques d'identification des relations. Notons que ce type d'exemples a motivé des travaux sur la construction de ressources liant paires de verbes et relations discursives (CONRATH et al., 2014). La simple présence de cette paire n'est cependant pas un indice suffisant comme nous avons pu le voir dans le cas où le discours est continué par un segment qui annule l'inférence d'une relation causale. De plus, le temps des verbes peut avoir une influence cruciale. Ainsi, pour le français, la succession de ces deux mêmes verbes au passé simple bloque l'inférence de la relation *Explanation* comme on peut le voir dans l'exemple (27b) présenté dans (BUSQUETS et al., 2001).

(27) a. Max est tombé. Paul l'a poussé.

b. Marie tomba. Jean la poussa.

En plus du temps des verbes, d'autres indices morphologiques peuvent aider à l'inférence. La présence du participe présent dans les exemples (28a) et (28b) guide ainsi l'interprétation d'une relation causale.

(28) a. [Les tours se sont effondrées moins de deux heures plus tard] [entraînant l'immeuble du Marriott World Trade Center dans leur chute.]

b. Paul a tiré sur Marie, la tuant.

D'autres indices, relevant de la sémantique lexicale, ont pu être identifiés. Par exemple, la relation *Elaboration* peut être inférée en SDRT si la seconde éventualité correspond à un sous-type de la première, si on reconnaît des hyperonymes. La polarité peut être un autre indice, permettant par exemple d'inférer *Contrast* en cas de polarité opposée dans les arguments, ou de la présence d'antonymes, comme l'opposition *nocturne/jour*, ou d'oppositions plus pragmatiques comme celles que l'on peut inférer entre *assert/prove*, *say/do* ou *clever/very clever* (JAYEZ et ROSSARI, 1998). On peut également se servir de la structure syntaxique pour signaler une relation, notamment des structures parallèles peuvent indiquer une comparaison. JAYEZ et ROSSARI (1998) citent également l'ordre des mots, l'inversion du sujet et du verbe pouvant indiquer une *Condition* (« *Savait-il*... »), et le mode des phrases, la relation *Solutionhood* peut par exemple être signalée par un mode interrogatif (TABOADA et DAS, 2013). La ponctuation peut également être un indice, il nous semble que l'utilisation des deux points, en lien avec des connaissances du monde, guide, dans l'exemple (29) issu d'ANNODIS, l'inférence d'une relation causale.

(29) [Tonneaux à Montreux-Vieux :]$_1$ [trois blessés graves]$_2$

La structure discursive elle-même peut guider l'interprétation, ainsi les relations de type *Summary* ont tendance à se trouver en fin de document. Il semble également qu'il existe des contraintes sur les enchaînements de relations, ROZE (2011) a ainsi étudié certaines de ces interactions à un niveau fin de l'analyse. Les systèmes d'analyse complets mettent par ailleurs parfois en jeu des mécanismes pour prendre en compte certaines de ces interactions, de façon cependant simplifiée. Les autres niveaux d'analyse au niveau du texte, coréférence, anaphore et structuration d'un texte en topique, participent tous de la cohésion d'un document et peuvent guider l'inférence d'une relation. Ainsi, en SDRT, certaines relations ne peuvent s'établir que si les segments mis en jeu correspondent au même topique, c'est le cas par exemple de la relation *Narration*. En ce qui concerne la coréférence, il est clair que les informations qu'elle fournit sont des indices de la relation *Entity Elaboration*, utilisée dans le corpus ANNODIS, ou de relations de la classe *Expansion* du PDTB. Elles peuvent également indiquer une relation de type *List*. Ces informations participent aussi de la cohésion des documents et sont donc liées à la question de la structuration en topiques. Il est probable qu'elles entrent également en compte dans l'identification d'autres relations, ou du moins qu'elles peuvent permettre de restreindre certaines inférences ou de rendre compte de certains phénomènes liés à la cohérence. Ainsi, si le discours en (30a), repris de (ASHER et LASCARIDES, 2003) est généralement considéré comme incohérent, celui en (30b) ne pose aucun problème de cohérence, la seule différence résidant dans le partage de sujet entre les éventualités décrites.

(30) a. *Pierre aime le sport. Il déteste le football.

b. Pierre aime le sport. Jean déteste le football.

Les études sur l'identification automatique des relations ont cherché à modéliser ces différents indices à l'aide de ressources variées. Cette diversité des informations à prendre en compte rend la tâche particulièrement difficile, puisqu'il faudrait idéalement être capable de faire des inférences complexes à partir de principes logiques et de connaissances encyclopédiques. Nous décrivons les systèmes automatiques et les indices qu'ils mettent en œuvre dans le chapitre suivant.

2.5 Description des données annotées utilisées

Dans cette section, nous décrivons les données utilisées dans cette thèse, c'est-à-dire les corpus manuellement annotés au niveau discursif, ANNODIS (2.5.1.1) et le PDTB (2.5.1.2), ainsi que les lexiques de connecteurs (2.5.2) qui nous servent à annoter automatiquement de nouvelles données dans les expériences décrites dans le chapitre 4.

2.5.1 Corpus manuellement annotés

2.5.1.1 ANNODIS

Nous avons déjà décrit précédemment la plupart des caractéristiques du corpus ANNODIS (AFANTENOS et al., 2012a). Nous donnons dans cette section des informations supplémentaires concernant les données annotées et précisons certains aspects de l'annotation. Rappelons que ce corpus s'inscrit dans le cadre général de la SDRT et que c'est, à ce jour, le seul corpus discursif entièrement annoté pour le français. C'est un corpus librement distribué [24].

24. http://redac.univ-tlse2.fr/corpus/annodis/

Le projet ANNODIS vise la construction d'un corpus annoté en discours pour le français à deux niveaux. La perspective dite ascendante correspond à la structuration d'un texte par des relations rhétoriques telles que décrites jusqu'à présent. La perspective dite descendante s'intéresse aux stratégies d'organisation fonctionnelle du document en particulier en ce qui concerne la continuité et la discontinuité textuelle et correspond à l'annotation des chaînes de topiques et des structures énumératives. Nous nous intéressons ici uniquement au niveau microstructurel correspondant à la perspective ascendante. La version du corpus utilisée (en date du 15/11/2012) comporte, pour ce niveau, 86 documents provenant de l'*Est Républicain* et de Wikipédia. Au total, 3 339 paires de segments liés à une ou plusieurs relations sont annotées, un nombre relativement faible si l'on compare au corpus du PDTB (environ 40 000 paires). Notons que certains documents sont annotés par des annotateurs « naïfs » et par des experts ce qui permet d'étudier les principes cognitifs de l'organisation discursive. Ainsi, ADAM et VERGEZ-COURET (2012) se sont intéressés à ces deux annotations pour les relations *Elaboration* et *Entity Elaboration* qui sont souvent confondues par les annotateurs naïfs et difficiles à identifier dans le cadre d'un système automatique car elles sont peu souvent marquées par un connecteur. Ces observations les ont amenés à définir de nouveaux indices pour ces relations fondés sur la cohésion lexicale entre les arguments.

Concernant la segmentation, le manuel d'annotation autorise une large diversité d'EDU et encourage les annotateurs à la sur-segmentation. Une segmentation erronée pourra être corrigée ensuite pendant la phase d'annotation des relations (MULLER et al., 2012b). Ce principe permet de rendre la segmentation relativement neutre et d'assurer la couverture la plus totale possible. Les EDU autorisées sont les propositions contenant la description d'(au moins) une éventualité : les phrases simples, les propositions formant une structure corrélative, les propositions principale et subordonnée dans le cas de structures conditionnelles, de subordonnées temporelles, concessives, causales, participiales, relatives, complétives et interrogatives indirectes introduites par un verbe assertif, informatif, de communication ou d'attitude propositionnelle. Nous avons vu qu'une spécificité de ce corpus concernait la segmentation de tous les adverbiaux quand ils sont détachés en tête de proposition, adverbiaux généralement arguments d'une relation *Frame*. Une autre particularité correspond à la segmentation de toutes les appositions, qu'elles soient contituées d'un participe passé ou présent, d'un adjectif ou d'un nom. Le manuel propose ainsi les exemples de segmentation (31a) et (31b) [25].

(31) a. [Saoule,] [elle est rentrée dans l'arbre avec sa voiture.]

b. [Mikhaïl Saakachvili,] [le jeune et bouillant président géorgien, avait besoin d'action pour sauver son régime.]

La phase d'attachement correspond à l'identification des segments à relier. Les relations sont annotées entre EDU ou entre CDU, contiguës ou non. L'attachement gouverne également l'ordre des arguments de la relation : en effet, ce n'est pas parce qu'un segment a se trouve dans le texte avant un segment b que la relation R prend ces segments en arguments dans cet ordre. Le segment attaché est en général le second argument de la relation [26]. Pour une proposition indépendante ou principale, on cherche son segment cible vers l'arrière (donc la seconde proposition sera attachée à la première, $R(a, b)$). Pour une subordonnée ou une apposition, elle s'attache à la principale quelque soit sa position ($R(a, b)$ ou $R(b, a)$). Pour les exemples explicites mettant en jeu un connecteur introduisant une proposition subordonnée, il est clair que l'on pourra avoir les ordres $R(a, b)$ — comme la relation *Explanation* $(1, 2)$ en (32a) — ou $R(b, a)$ — comme la relation *Explanation* $(2, 1)$

25. La position du RST DT n'est pas très claire sur ce point, le manuel indique de segmenter les appositions mais, contrairement à d'autres catégories, ne précise pas si l'apposition doit contenir un verbe (le seul exemple donné en contient un). Les principes d'annotation du PDTB ne prévoient pas, quant à eux, une telle segmentation.

26. Il y a des exceptions, par exemple pour la relation *Conditional*, la subordonnée conditionnelle correspond au premier argument de la relation.

en (32b). Avec des propositions indépendantes, la règle d'attachement décide en partie du type de relation annoté. Par exemple, l'exemple implicite (32c) correspond à *Explanation* $(1, 2)$ — parce que la cause 2 est attachée (principale vers l'arrière) à l'effet 1. Si les phrases étaient inversées comme en (32d), on aurait une relation *Result* $(1, 2)$ — parce que l'effet 2 est attaché à la cause 1. Mais il est également possible d'avoir une relation *Explanation* implicite avec les arguments inversés, comme en (32e), exemple provenant du corpus : la cause 1 est attachée à l'effet 2, on a donc *Explanation* $(2, 1)$. Les systèmes dédiés à l'identification automatique des relations, hors d'un système complet, ignorent la phase d'attachement, et considère donc une paire de segments que l'on sait attachés et pour lesquels on connaît l'ordre. Concernant l'ordre, notons que le problème ne concerne que les propositions subordonnées et qu'il se fonde ici sur des critères syntaxiques, il correspond donc, nous semble-t-il, à une information relativement simple à récupérer.

(32) a. [Ève est heureuse]$_1$ [parce que le permis de construire est signé.]$_2$
b. [Comme le permis de construire est signé,]$_1$ [Ève est heureuse.]$_2$
c. [L'équipe a perdu lamentablement hier.]$_1$ [Elle avait trop de blessés.]$_2$
d. [L'équipe avait trop de blessés.]$_1$ [Elle a perdu lamentablement (hier).]$_2$
e. [Ceci étant un crime pour les Turco-Mongols,]$_1$ [le Grand Émir parcourt les centaines de kilomètres qui le séparent d'Ispahan]$_2$

Le guide d'annotation donne des descriptions intuitives des relations, quelques exemples et quelques marqueurs potentiels mais sans fournir une liste exhaustive. Les relations choisies sont décrites comme plus ou moins communes à divers cadres théoriques avec un grain moyen : on a ainsi 18 relations (rhétoriques) annotées, un nombre proche du nombre de relations dans le PDTB et des 16 groupes du RST DT. Parmi ces relations, une seule méta-relation, *Explanation* *, est annotée même si les annotateurs pouvaient en proposer d'autres. De plus, cette relation ne correspond qu'à 8 exemples. Ce faible nombre d'annotations n'est probablement pas dû à leur faible représentation dans les données mais plutôt au fait qu'elles sont particulièrement difficiles à identifier, c'est-à-dire à séparer de leur relation sémantique correspondante. Nous avons dit qu'une relation est dédiée à l'identification des phénomènes d'attribution, comme dans le RST DT, et que les relations *Frame* et *Temploc*, n'appartenant pas à la SDRT, avaient été ajoutées. Une relation supplémentaire, non rhétorique, permet de fusionner des EDU après l'étape de segmentation.

Le nombre d'exemples par relation dans le corpus est donné dans le tableau 2.1, nous indiquons aussi les groupements proposés pour ces relations [27]. Dans nos expériences, nous utilisons un sous-ensemble de relations que nous présentons dans la section 4.4.1. Le tableau 2.1 montre un grand déséquilibre entre les différentes relations, la relation la plus représentée (*Continuation*) correspond à environ 20% des données et la moitié des relations est représentée par moins de 100 exemples. En plus de la relation *Continuation*, les relations les plus représentées sont celles du groupe des élaborations, des relations de sémantique faible. Nous verrons ci-dessous une situation similaire dans le PDTB où le groupe le plus représenté, les relations de type *Expansion*, correspondent aussi globalement à des relations de sémantique faible.

Afantenos et al. (2012a) rapportent un score d'accord inter-annotateur pour la tâche d'attachement de 66% de F_1 et, pour la tâche d'étiquetage en relation, un score Kappa de $0,4$ en considérant l'ensemble de relations et de $0,57$ en ne considérant que la division entre relations coordonnantes et subordonnantes. Par comparaison, pour des annotateurs entraînés, Carlson et al. (2003) rapportent un score d'accord inter-annotateur kappa relativement haut, $0,72$, pour l'identification des relations parmi 78 relations possibles et de $0,77$ pour les relations regroupées en 16 classes.

27. Afantenos et al. (2012a) rapportent des chiffres légèrement différents et ne rapportent pas de nombre d'exemples pour les relations *Conditional* et *Explanation**.

Groupe	Relation	# exemples	Fréquence relative
Causalité	*Explanation*	120	3,6%
	Goal	94	2,8%
	Result	162	4,8%
Structurel	*Parralel*	59	1,8%
	Contrast	142	4,2%
	Continuation	676	20,2%
Logique	*Alternation*	18	0,5%
	Conditional	20	0,6%
Discours rapporté	*Attribution*	74	2,2%
Exposition/Narration	*Background*	155	4,6%
	Narration	349	10,4%
	Flashback	27	0,8%
	Frame	207	6,2%
	Temploc	18	0,5%
Élaborations	*Elaboration*	611	18,3%
	Entity Elaboration	524	15,7%
Commentaire	*Comment*	75	2,2%
Méta-relations	*Explanation**	8	0,2%
Total		3 339	

Table 2.1.: Corpus ANNODIS : nombre d'exemples (« # exemples ») par relation et fréquence relative dans la version du corpus utilisée. Ces chiffres ne prennent pas en compte les annotations multiples (AFANTENOS et al., 2012a).

2.5.1.2 *Penn Discourse Treebank*

Le PDTB (PRASAD et al., 2008a) est construit comme une surcouche d'annotation sur le *Penn Treebank* (MARCUS et al., 1993) correspondant à une annotation syntaxique manuelle, et qui a par ailleurs été annoté à d'autres niveaux dans les projets PropBank (KINGSBURY et PALMER, 2002) et OntoNotes (HOVY et al., 2006). Il est donc uniquement constitué d'articles journalistiques issus du *Wall Street Journal*. Par rapport aux autres corpus discursifs, comme le RST DT ou ANNODIS, il présente la particularité de ne pas chercher à établir une structure globale couvrant tout le document, de fonder le processus d'annotation sur les connecteurs et de proposer une organisation hiérarchique des relations autorisant des sous-spécifications. Finalement, le corpus contient 2 259 documents et l'annotation de 40 600 paires de segments éventuellement annotés avec plus d'une relation.

Rappelons que le processus d'annotation commence par l'identification des connecteurs. Ensuite, les annotateurs doivent identifier leurs arguments et la ou les relations qu'ils déclenchent, appelées sens du connecteur. L'étape finale consiste à annoter une relation entre toutes les phrases adjacentes au sein d'un même paragraphe non liées par un connecteur (explicite). Cette dernière phase correspond à l'annotation des relations non explicites parmi lesquelles les auteurs ont distingué relations implicites, lexicalisations alternatives (AltLex) et relations d'entité (EntRel). Enfin, une étiquette supplémentaire permet d'indiquer l'absence de relation entre les phrases considérées (NoRel). Comme nous l'avons dit, l'annotation n'est pas complète à ce jour puisqu'il manque entre autres les relations implicites intra-phrastiques, entre paragraphes et entre phrases non adjacentes ainsi que les relations implicites déclenchées entre des segments liés par des relations explicites.

Notons ici que sont aussi annotés les adverbes modifieurs de connecteurs comme *only*, *even* ou *at least*. Plus précisément, dans les données, un exemple est associé à ce qui est appelé la tête du connecteur, c'est-à-dire le connecteur non modifié, et à une forme dite brute du connecteur, contenant éventuellement le modifieur. Ainsi, dans l'exemple de relation explicite (33) provenant

du manuel d'annotation (PRASAD et al., 2007), le connecteur identifié est *because* et sa forme brute contient le modifieur « *partly* ».

(33) [We're seeing it] (partly) *because* $_{\text{explicit}}$ [older vintages are growing more scarce.]

Concernant la question de l'emploi discursif des formes, correspondant dans les systèmes automatiques à la tâche d'identification des connecteurs, le manuel d'annotation reste relativement flou. Le premier guide d'annotation (PRASAD et al., 2006) indique bien que l'on peut trouver dans les données des formes « homonymes » à des expressions considérées comme connecteur mais qui n'apportent pas de contribution au niveau discursif, ces formes n'étant bien sûr pas annotées comme connecteur. Cependant, sont seulement évoqués les cas de formes ne liant pas des segments discursifs (*and* coordonnant des groupes nominaux et *for example* modifiant un groupe nominal) et de formes ne modifiant pas la clause globale mais seulement un adjoint. De plus, nous n'avons pas trouvé de scores d'accord inter-annotateur sur cette étape. Comme la liste de connecteurs a été constituée en partant de listes pré-établies mais aussi en considérant les données annotées, il semble que cette étape ne corresponde pas dans ce corpus à une annotation au sens strict, partant de règles pré-définies et menée en double aveugle. Notons finalement que le nombre de connecteurs pris en compte est relativement bas, seulement 100 tandis que l'inventaire de KNOTT (1997) compte 350 formes [28].

La seconde phase d'annotation correspond à l'identification des arguments des connecteurs. Cette tâche peut être divisée en deux étapes : trouver la localisation des arguments par rapport au connecteur et déterminer leur empan. Dans le PDTB, les arguments sont étiquetés de manière neutre par *Arg1* et *Arg2*, les chiffres référant à l'ordre des arguments de la relation $R(\text{Arg1}, \text{Arg2})$. L'étiquette *Arg2* identifie l'argument syntaxiquement attaché au connecteur, *Arg1* correspond à l'autre argument. Comme pour ANNODIS, l'ordre des arguments pour un exemple explicite dépend donc de la réalisation en syntaxe : le segment contenant le connecteur est toujours le second argument, qu'il se positionne avant ou après le premier dans le texte. Notons que pour les relations implicites seulement annotées entre phrases adjacentes, un connecteur est inséré entre les deux phrases et marque le début de l'*Arg2* : les arguments suivent donc l'ordre du texte. L'*Arg2* d'une relation explicite est donc facile à localiser, le problème se pose plutôt pour l'*Arg1*. Les arguments des relations explicites ne sont pas forcément adjacents, l'*Arg1* d'un connecteur peut se trouver dans la même phrase que le connecteur (60.9% des cas), dans une phrase précédente adjacente (30.1%) ou non (9%) ou dans une phrase suivante, ce dernier cas ne regroupant cependant que 8 exemples (PRASAD et al., 2008a). Dans le cas où les deux arguments sont dans la même phrase, les positions relatives des deux arguments diffèrent selon la catégorie du connecteur, comme nous avons pu le voir en section 2.4.1.

Le manuel indique de plus le type d'argument accepté. Comme pour les autres corpus, la proposition est toujours acceptée, qu'elle soit principale, complément ou subordonnée. Les syntagmes verbaux coordonnés sont acceptés à condition qu'ils soient arguments d'un connecteur autre que la coordination elle-même (qui n'est pas annotée comme connecteur dans ce cas). Un exemple d'une telle situation est donné en (34a), on rappelle que les connecteurs ne sont pas inclus dans les arguments, on a donc ici un *Arg2* discontinu qui contient « *and* » ainsi que le syntagme verbal « *immediately sued the Bell Co.* ». Nous avons déjà évoqué le cas des syntagmes nominaux annotés si l'*Arg1* peut être interprété de manière existentielle, c'est aussi le cas s'il contient un « cas clairement observable » de nominalisation comme dans l'exemple (34b). Enfin, les expressions anaphoriques comme « *this* » ou « *so* » peuvent être considérées comme des arguments ainsi que les réponses à

28. Notons que pour le FDTB1 (STEINLIN et al., 2015) correspondant à une annotation plus exhaustive des formes correspondant à des connecteurs, cette seule étape a pris deux ans.

des questions comme « *yes* » dans l'exemple (34c). Ces deux derniers points impliquent que des mécanismes de récupération d'informations contextuelles sont nécessaires pour obtenir un contenu pour certains arguments.

(34) a. [It acquired Thomas Edison's microphone patent]$_1$ [and]$_2$ *then* [immediately sued the Bell Co.]$_2$

b. But in 1976, the court permitted [resurrection of such laws,]$_1$ *if* [they meet certain procedural requirements.]$_2$

c. Underclass youth are a special concern. ($_{Sup1}$ Are such expenditures worthwhile, then) ? [Yes]$_1$, *if* [targeted.]$_2$

Une fois des arguments valides localisés, il faut en déterminer l'empan. Un argument peut éventuellement dépasser le cadre de la phrase. De plus, le corpus présente la particularité d'introduire le principe de minimalité que nous avons évoqué : les annotateurs ne doivent inclure dans l'argument que le contenu minimalement nécessaire à l'inférence de la relation, éventuellement donc en mettant en jeu un système de résolution d'anaphore. Tout segment textuel perçu comme non nécessaire mais pertinent est annoté comme matériel supplémentaire à l'aide des étiquettes *Sup1* pour l'*Arg1* et *Sup2* pour l'*Arg2*. Les arguments doivent cependant contenir des propositions complètes. Des conventions sont détaillées afin de déterminer ce qui doit être considéré comme nécessaire. Cette annotation peut permettre d'inclure des informations sur les référents d'anaphores ou les annotations de question-réponse comme dans l'exemple (34c). Ce principe de minimalité s'applique aussi pour déterminer les arguments des relations implicites et des cas de lexicalisation alternative. L'annotation d'informations supplémentaires n'est probablement pas consistante dans le corpus, PRASAD et al. (2014) notent ainsi que ces informations ont été sous-annotées et qu'une étude doit être menée pour comprendre s'il s'agit réellement d'une caractéristique motivée des relations discursives. Notons que seules $8,51\%$ des relations explicites et $0,78\%$ des relations implicites contiennent l'annotation d'un *Sup1* ou d'un *Sup2*. Ce concept, qui participe de la couverture non totale du corpus, semble de plus assez subjectif et difficile à appréhender, il est abandonné par exemple dans le corpus français suivant le modèle d'annotation du PDTB (DANLOS et al., 2012). Pour l'étape d'identification des arguments, PRASAD et al. (2008a) donnent un score d'accord inter-annotateur de $90,2\%$ pour les exemples explicites et de $85,1\%$ pour les implicites en correspondance exacte, et de $95,4\%$ et $92,6\%$ respectivement en correspondance partielle.

Comme nous l'avons dit, les sens ou relations du PDTB sont organisés en une hiérarchie à trois niveaux comportant 4 classes, 16 types et 23 sous-types. Le guide d'annotation fournit des descriptions intuitives des relations, parfois légèrement formalisées, et des exemples. Par contre, il ne propose pas une correspondance même partielle entre les connecteurs inventoriés et les relations qu'ils peuvent potentiellement déclencher : c'est l'annotation qui va révéler l'ambiguïté en relation des connecteurs. Nous représentons la hiérarchie de sens dans le tableau 2.2. Chaque classe est divisée en plusieurs types, mais un type ne correspond pas forcément à plusieurs sous-types. Les annotateurs peuvent annoter un exemple en allant au plus profond de la hiérarchie, par exemple avec le sens *Temporal.Asynchronous.Precedence*, ou seulement avec une classe, *Temporal*, ou un type, *Temporal.Asynchronous*. Ceci permet de gérer le doute, en autorisant des sous-spécifications, et de faciliter l'annotation. Il est aussi clair que l'accord inter-annotateur est meilleur pour des relations de grain moins fin et notamment pour les 4 classes. Ainsi, PRASAD et al. (2008a) rapportent un accord de 94% pour les classes, de 84% pour les types et de 80% pour les sous-types. Ces chiffres prennent en compte les relations explicites, implicites et AltLex, et ne rendent donc pas forcément compte d'une difficulté potentiellement plus importante pour les implicites.

Une autre particularité de l'annotation est d'autoriser des annotations multiples. Un exemple explicite ou AltLex peut ainsi correspondre à deux sens. Un exemple implicite peut être annoté avec deux connecteurs, chacun pouvant déclencher jusqu'à deux sens. Ainsi, dans l'exemple (35), l'annotateur a inséré deux connecteurs « implicites » entre les phrases, *because* et *for example*.

(35) [The third principal in the S. Gardens adventure did have garden experience.]$_1$ *because* $_{\text{Implicit}}$ *for example* $_{\text{Implicit}}$ [The firm of Bruce Kelly/David Varnell Landscape Architects had created Central Park's Strawberry Fields and Shakespeare Garden.]$_2$

Ce phénomène est cependant relativement peu répandu, seul $5,4\%$ des exemples explicites sont annotés avec deux sens, $2,2\%$ des implicites correspondent à un connecteur annoté avec deux sens, 1.1% à deux connecteurs chacun avec un sens, et 1 exemple contient deux connecteurs annotés avec deux sens. Une étude plus détaillée sur ce point peut être trouvée dans (PRASAD et al., 2014). Comme nous le verrons, les systèmes automatiques gèrent différemment cette question, soit en dupliquant les exemples pour les différentes relations, ce qui pose problème pour un système statistique, soit en ne conservant que la première annotation.

La dernière couche d'annotation du corpus correspond à l'attribution pour les exemples explicites, implicites et AltLex. Nous avons détaillé cette annotation dans la section 2.3.5.1. Cette information n'a pas, à notre connaissance, été utilisée pour l'identification automatique des relations.

Nous donnons dans la table 2.2 la distribution des sens pour ces trois types d'exemples en ne considérant que la première relation annotée. Pour chaque niveau, nous indiquons le nombre d'exemples disponibles, ce qui signifie, par exemple, qu'un exemple annoté au niveau 2 ou 3 entre également dans le nombre d'exemples annotés au niveau 1. Par contre, un exemple annoté uniquement au niveau 1 n'entre pas dans les comptes aux niveaux inférieurs, ce qui explique que les totaux soient différents pour chaque niveau. Le corpus contient en plus $5\ 210$ EntRel et 254 NoRel.

Les fréquences relatives présentées dans ce tableau montrent que les données annotées dans ce corpus sont fortement déséquilibrées : certaines relations apparaissent avec des fréquences particulièrement basses, et le corpus est dominé par un petit ensemble de relations dépendant du type de données pris en compte (entre explicite, implicite et AltLex). Notons que les données sont particulièrement déséquilibrées pour les exemples de type implicite, avec, pour le premier niveau de hiérarchie, $53,6\%$ des données annotées avec la relation *Expansion* contre seulement $5,1\%$ pour la relation *Temporal*, ce qui tend à montrer, si l'on compare aux chiffres pour le type explicite ($18,6\%$ de *Temporal*), une préférence pour l'explicitation des liens temporels. Au contraire, la classe *Contingency* est la moins représentée parmi les relations explicites, tandis qu'elle est la seconde plus représentée parmi les implicites : il semble donc que cette classe, et notamment les liens causaux, soit plus facilement exprimée de manière implicite.

Ce tableau montre également que certaines relations ne sont jamais exprimées de manière implicite (du moins avec les limitations actuelles du corpus, c'est-à-dire la restriction aux liens implicites interphrastiques et intraparagraphes). Ainsi les types de deuxième niveau *Condition* et *Pragmatic Condition* ne correspondent qu'à un seul exemple implicite chacun. On note que la situation est similaire avec le type AltLex. Cependant, s'il nous semble effectivement difficile d'exprimer une relation conditionnelle de manière implicite, des lexicalisations alternatives des connecteurs semblent par contre tout à fait envisageables (par exemple un verbe comme « suppose »). Il est donc probable que les proportions d'AltLex pour ces relations augmenteront avec l'annotation des relations intra-phrastiques non explicites. De manière générale, le nombre de relations de type AltLex est assez bas, ce qui rend cette classe difficile à étudier. Rappelons que certaines formes

Relations	Explicit		Implicit		AltLex	
	#	f	#	f	#	f
Temporal	3 440	18, 6%	826	5, 1%	86	13, 7%
Asynchronous	2 022	11, 2%	650	4, 1%	67	10, 8%
Precedence	950	11, 2%	499	5, 6%	48	11, 1%
Succession	1 069	12, 6%	151	1, 7%	19	4, 4%
Synchronous	1 413	7, 8%	175	1, 1%	19	3, 1%
Contingency	3 250	17, 6%	4 185	26, 1%	275	44, 1%
Cause	1 818	10, 1%	4 113	26, 0%	272	43, 8%
Reason	1 201	14, 2%	2 434	16, 0%	101	23, 4%
Result	617	7, 3%	1 678	18, 7%	171	39, 6%
Pragmatic Cause	8	$< 0, 1\%$	69	0, 4%	1	0, 2%
Justification	8	0, 1%	69	0, 8%	1	0, 2%
Condition	1 356	7, 5%	1	$< 0, 1\%$	2	0, 3%
Hypothetical	751	8, 9%	1	$< 0, 1\%$	0	0%
General	327	3, 9%	0	0%	2	0, 5%
Unreal Present	123	1, 4%	0	0%	0	0%
Unreal Past	54	0, 6%	0	0%	0	0%
Factual Present	90	1, 1%	0	0%	0	0%
Factual Past	9	0, 1%	0	0%	0	0%
Pragmatic Condition	67	0, 4%	1	$< 0, 1\%$	0	0%
Relevance	21	0, 3%	1	$< 0, 1\%$	0	0%
Implicit Assertion	46	0, 5%	0	0%	0	0%
Comparison	5 471	29, 6%	2 441	15, 2%	46	7, 4%
Contrast	3 844	21, 3%	2 063	13, 0%	40	6, 4%
Juxtaposition	1 140	13, 4%	700	7, 8%	27	6, 2%
Opposition	347	4, 1%	141	1, 6%	1	0, 2%
Pragmatic Contrast	18	0, 1%	2	$< 0, 1\%$	1	0, 2%
Concession	1 201	6, 7%	219	1, 4%	5	0, 2%
Expectation	386	4, 5%	31	0, 3%	1	0, 2%
Contra-expectation	798	9, 4%	182	2, 0%	4	0, 9%
Pragmatic Concession	11	0, 1%	1	$< 0, 1\%$	0	0%
Expansion	6 298	34, 1%	8 601	53, 6%	217	34, 8%
Conjunction	5 212	28, 9%	3 440	21, 8%	111	17, 9%
Instantiation	302	1, 7%	1 395	8, 8%	38	6, 1%
Restatement	155	0, 9%	3 108	19, 7%	63	10, 1%
Specification	109	1, 3%	2 433	27, 1%	40	9, 3%
Equivalence	12	0, 1%	273	3, 0%	5	1, 2%
Generalization	17	0, 2%	190	2, 1%	12	2, 8%
Alternative	351	1, 9%	180	1, 1%	0	0%
Conjunctive	47	0, 5%	10	0, 1%	0	0%
Disjunctive	143	1, 7%	0	0%	0	0%
Chosen Alternative	115	1, 4%	167	1, 9%	0	0%
Exception	14	0, 1%	1	$< 0, 1\%$	1	0, 2%
List	240	1, 3%	386	2, 4%	1	0, 2%
Total Niveau 1	18 459		16 053		624	
Total Niveau 2	18 032		15 804		621	
Total Niveau 3	8 379		8 960		432	

Table 2.2.: Corpus PDTB : nombre d'exemples par relation (« # ») et fréquence relative (« f ») par rapport au total d'exemples de même niveau et de même type. Pour le niveau 2 (resp. 1) le nombre d'exemples prend en compte les exemples annotés au niveau 3 (resp. 2 et 3). Ces chiffres sont repris du manuel d'annotation (PRASAD et al., 2007), les annexes fournissent des indications du nombre de relations annotées, nous ignorons simplement les annotations multiples.

annotées comme AltLex auraient dû être considérées comme des connecteurs car ils en remplissent les critères définitoires (PRASAD et al., 2014). Par ailleurs, PRASAD et al. (2014) précisent également que l'annotation de ces formes n'a pas été faite de manière exhaustive, la reconnaissance d'une forme alternative étant laissée à la subjectivité des annotateurs : certaines relations de type AltLex doivent donc être contenues dans les données implicites.

2.5.2 Lexiques de connecteurs

Comme nous l'avons dit, on considère généralement que les connecteurs appartiennent à une classe plus ou moins fermée. Étant donné qu'ils constituent des signaux forts de relations discursives, différents travaux ont été menés pour en constituer des inventaires ou lexiques de connecteurs. Ces inventaires peuvent être utilisés pour aider à l'annotation de corpus, nous l'avons vu, des listes de connecteurs ont été ainsi fournies aux annotateurs dans le cadre des projets ANNODIS ou RST DT (MULLER et al., 2012b ; CARLSON et MARCU, 2001). Ils peuvent également être utiles dans le cadre de systèmes de génération automatique de texte (MARCU, 1997b ; STEDE et UMBACH, 1998 ; DANLOS, 2000 ; PRASAD et al., 2005). Enfin, ils constituent évidemment des traits cruciaux dans le cadre de systèmes automatiques des relations discursives (explicites) (PITLER et al., 2008 ; PITLER et NENKOVA, 2009 ; LIN et al., 2010). Dans le cadre de cette thèse, nous utilisons les connecteurs discursifs pour annoter automatiquement des relations explicites (chapitre 4), suivant l'idée d'abord proposée par (MARCU et ECHIHABI, 2002), et pour construire une représentation de mots distributionnelle en anglais mettant en jeu une dimension rhétorique (chapitre 5). Nous décrivons donc ici les inventaires de connecteurs utilisés dans cette thèse, à savoir celui du PDTB et le lexique des connecteurs du français, ou LexConn.

2.5.2.1 Inventaire du PDTB

Dans le PDTB, les auteurs ont défini un ensemble restreint de connecteurs en se fondant sur des listes établies dans des études précédentes comme (KNOTT, 1997) et (FORBES-RILEY et WEBBER, 2006), éventuellement élargies en tenant compte des données en cours d'annotation. Ils ont exclu les marqueurs signalant un changement de topique identifiés dans (HIRSCHBERG et LITMAN, 1987). Devaient aussi être normalement exclus les marqueurs pragmatiques, comme *actually* et *in fact* qui indiquent que la clause dans laquelle ils apparaissent a une fonction conversationnelle, alimentant ou contredisant un énoncé prononcé précédemment. Il s'avère que *in fact* a finalement été annoté, PRASAD et al. (2014) notent que cette annotation « accidentelle » fournit cependant des informations intéressantes sur ce type particulier de connecteur. Enfin, les expressions contenant une forme d'anaphore, comme *for that reason*, n'ont pas été considérées non plus comme des connecteurs, l'argument étant ici que cette annotation devrait attendre l'annotation des liens de coréférence.

Certaines de ces formes ont cependant été prises en compte dans l'annotation des relations de type lexicalisation alternative. Cette catégorie est associée aux exemples pour lesquels l'insertion d'un connecteur paraît introduire une forme de redondance. Dans ce cas, les annotateurs devaient indiquer l'expression permettant d'inférer la relation. Ces expressions peuvent être de catégories différentes des connecteurs discursifs, incluant par exemple des choses comme *one reason is*, qui n'est pas une expression figée, mais aussi des expressions comme *quite the contrary, eventually* et *thereafter*, ce dernier comptant pour environ 15% des données, qui pourraient toutes être considérées comme des connecteurs discursifs. La catégorie des lexicalisations alternatives est donc très hétérogène, certaines expressions ayant de plus été considérées comme telles par une partie des annotateurs mais pas par tous (comme *that means* par exemple). PRASAD et al. (2014) concluent sur l'impossibilité d'utiliser ces lexicalisations dans une approche d'apprentissage automatique.

Finalement, 100 connecteurs ont été annotés. Ils sont listés dans (PRASAD et al., 2007), nous les reprenons en annexe A. Il semble cependant que le projet prévoyait au début une annotation de 250 formes (MILTSAKAKI et al., 2004), la raison d'une limitation dans la version finale à 100 formes n'étant pas expliquée. Il est possible que de futures annotations viennent enrichir l'inventaire actuel. Les possibilités d'ambiguïté en emploi et en relation ne sont pas prédéfinies, c'est l'annotation du corpus qui a permis de les révéler.

2.5.2.2 Lexique des connecteurs du français

Le LexConn (ROZE et al., 2012) est également construit sur des listes existantes, provenant de lexiques français mais aussi de l'inventaire de (KNOTT, 1997) traduit manuellement. La liste de connecteurs candidats ainsi constituée est ensuite filtrée en se fondant sur différents critères syntaxiques, sémantiques et discursifs définis à partir de différentes propriétés des connecteurs. Ainsi, le critère de clivage stipule qu'un connecteur ne peut pas être focalisé dans une clivée, et les critères de substitution et de compositionnalité qu'un connecteur n'est pas une expression anaphorique — on ne peut substituer l'une de ses parties par une entité du contexte —, et n'est pas compositionnel — il est invariable et ne peut subir de modification interne. D'autres critères concernent la capacité d'un connecteur à déclencher une relation, à rendre un discours cohérent ou au contraire à le rendre incohérent.

Ces différents critères permettent par exemple de considérer comme connecteur l'expression *à part ça* et à rejeter les expressions « *comme ça* » et « *après ça* », distinction qui se fonde sur l'emploi, anaphorique ou non, de « *ça* ». Le critère de compositionnalité permet de distinguer différentes expressions contenant un substantif : les expressions qui contiennent un substantif qui ne peut être fléchi ou modifié et/ou un déterminant ne pouvant varier sont conservées. Ainsi, une expression comme *la preuve* répond à ces contraintes (elle ne peut être remplacée par « *les preuves* » ou « *une preuve* »). De même, *en tout cas* peut être un connecteur car on ne peut pas le modifier (« *en tout cas envisagé* »). Ces critères ont aussi mis en lumière les ambiguïtés en emploi de certains connecteurs. Le critère de clivage révèle par exemple que *ainsi* peut fonctionner comme connecteur discursif quand il est en position initiale, une information encodée dans le lexique. Ils révèlent également certaines propriétés des connecteurs et des relations qu'ils déclenchent. Ainsi, le critère de cohérence rend compte de la différence entre les discours (36a) et (36b), différence qui reflète la nécessité de l'usage d'un connecteur pour déclencher le type de contraste exprimé ici.

(36) a. ∗Marie est intelligente. Elle dit souvent des bêtises.

b. Marie est intelligente. *Quoiqu'* elle dit souvent des bêtises.

Une fois les connecteurs identifiés, ROZE et al. (2012) ont déterminé les relations qu'ils pouvaient déclencher en se fondant sur des données issues du corpus FRANTEXT. Les auteurs ont considéré 15 relations issues de l'inventaire de la SDRT. Les tests définis pour identifier la ou les relations discursives déclenchées par un connecteur se fondent sur le caractère subordonnant ou coordonnant des relations en testant les possibilités d'attachement des segments, sur la possibilité de substitution d'un connecteur par un autre et sur les effets sémantiques des relations.

L'inventaire contient finalement 328 connecteurs correspondant à 428 paires connecteur-relation [29]. Les connecteurs utilisés pour construire le corpus artificiel sur le français, décrit en section 4.4.1.2, sont listés en annexe B. Chaque entrée de Lexconn indique de plus la ou les formes que peut

29. La version utilisée dans cette thèse est téléchargeable sur la page `http://www.linguist.univ-paris-diderot.fr/~croze/`. La version utilisée dans le cadre du FDTB est téléchargeable sur la page `https://gforge.inria.fr/frs/?group_id=6145`, cette version a été augmentée suite à l'annotation des données.

prendre le connecteur, sa catégorie morpho-syntaxique, le caractère subordonnant ou coordonnant de la relation et éventuellement la position de la clause hôte par rapport au segment convié dans le cas d'une conjonction de subordination et, dans le cas d'un adverbe, la position du connecteur au sein de sa clause hôte. Des exemples et des commentaires viennent enrichir la base. Les connecteurs de type adverbial sont les plus fréquents avec 206 formes représentées. Viennent ensuite les conjonctions de subordination (174), les prépositions (36) et les conjonctions de coordination (12).

3 Identification automatique des relations implicites

Sommaire

Le domaine de l'analyse discursive automatique est relativement récent en TAL, le premier système datant de 1999 (MARCU, 1999). Il est cependant devenu un champ très actif avec l'annotation de nouveaux corpus, augmentant la masse de données annotées disponibles, et l'utilisation d'informations discursives pour améliorer d'autres systèmes de TAL comme la génération automatique (TABOADA et MANN, 2006a), le résumé automatique (DAUMÉ III et MARCU, 2009 ; THIONE et al., 2004 ; SPORLEDER et LAPATA, 2005), les systèmes question-réponse (CHAI et JIN, 2004 ; VERBERNE, 2007), l'évaluation de la qualité de devoirs (BURSTEIN et al., 2003 ; HIGGINS et al., 2004) ou l'analyse de sentiment (POLANYI et ZAENEN, 2006 ; VOLL et TABOADA, 2007 ; BHATIA et al., 2015). De plus, notons qu'une première tâche partagée a été organisée cette année dans le cadre de la conférence CoNLL [1] (*Conference on Natural Language Learning*), les participants étant invités à construire des systèmes d'analyse discursive complets à partir de données du type de celles annotées dans le PDTB. Cet évènement va permettre, comme ça a été le cas pour d'autres niveaux d'analyse, de définir les performances actuelles sur la tâche et de comparer ou d'innover au niveau des méthodes employées.

Un analyseur discursif automatique construit la structure discursive d'un document. On peut identifier de manière générale plusieurs tâches au sein d'un tel système comme la segmentation du texte en EDU, l'attachement des segments — c'est-à-dire l'identification de la structure, sans étiquette

1. http://www.cs.brandeis.edu/~clp/conll15st/

de relation — et l'identification des relations discursives les liant. Les systèmes automatiques, complets ou dédiés aux sous-tâches, sont relativement dépendants des cadres sur lesquels reposent les corpus. En particulier, la construction de la structure dépend fortement, au niveau du processus d'attachement, du cadre théorique et des contraintes qu'il définit liées au type de structure postulé, arbre pour le RST DT, graphe pour ANNODIS ou structure partielle pour le PDTB. Les systèmes de segmentation dépendent plus faiblement du cadre puisque, comme nous l'avons vu, les principes de segmentation sont relativement consensuels. Le PDTB fait cependant quelque peu exception puisque la segmentation n'est pas la première étape et correspond quelque part à des hypothèses simplificatrices : si tout système de segmentation prend en compte les connecteurs discursifs, qui sont de très bons indices, un tel système construit sur le PDTB ne nécessite pas l'identification des EDU intra-phrastiques ne contenant pas de connecteur ce qui simplifie largement la tâche. Enfin, l'identification automatique des relations discursives est liée au cadre théorique qui définit la nature des segments liés et le jeu de relations à identifier mais sans dépendance forte puisque tous les cadres aboutissent plus ou moins à des segments de nature analogue et des relations similaires. L'annotation du PDTB a permis de séparer cette tâche en deux sous-tâches : l'identification des relations explicites et celle des relations implicites. En particulier, les systèmes construits à partir de ce corpus ont démontré la difficulté de cette identification pour les relations implicites. Étant donné que ces relations représentent une bonne moitié des données, il est clair que les basses performances actuelles sur cette tâche ont un impact majeur sur les systèmes complets et donc sur la possibilité de les utiliser pour améliorer d'autres tâches de TAL.

Nous présentons dans la section suivante 3.1 les systèmes complets d'analyse discursive en rapportant les scores obtenus par ces systèmes pour les différentes tâches et pour l'analyse complète. Nous séparons les systèmes construits sur des corpus se fondant sur les cadres de la SDRT ou de la RST, pour lesquels nous réservons le terme d'*analyseur discursif* (section 3.1.1), des systèmes construits sur le PDTB que nous appelons *chunkers discursifs* (section 3.1.2). La grande majorité de ces systèmes ont été construits pour l'anglais. Les analyseurs construits dans le cadre de la RST ont été développés à partir du corpus RST DT, donc pour la langue anglaise. Nous n'avons pas connaissance d'analyseurs développés à partir d'un corpus construit sur le modèle du PDTB mais sur une langue autre que l'anglais. Pour la SDRT, par contre, à notre connaissance, seul le corpus français ANNODIS a donné lieu au développement d'un analyseur complet. Nous présentons ensuite (section 3.1.3) les applications de TAL utilisant des informations discursives dont la diversité reflète l'importance de la construction de systèmes automatiques performants. Les sections suivantes sont consacrées à la tâche d'identification automatique des relations implicites dont nous précisons le caractère crucial décrivons les difficultés en section 3.2. Nous présentons ensuite les études existantes sur cette tâche en section 3.3. Finalement, nous donnons des résultats pour des systèmes de référence en section 3.4, résultats qui pourront servir de référence tout au long de cette thèse.

3.1 Analyse automatique du discours

Dans cette section, nous décrivons les systèmes d'analyse discursive existants en nous focalisant sur les systèmes développés sur les corpus RST DT, ANNODIS et PDTB sur lesquels ont été construits la majorité des systèmes existants. Nous avons décrit dans le chapitre précédent les différences entre ces corpus, différences qui se reflètent dans les systèmes automatiques construits. Notamment, les systèmes reproduisent le processus d'annotation, on a donc clairement une différence entre ceux construits à partir de corpus comme le RST DT ou ANNODIS et ceux utilisant les données du PDTB. Dans le premier cas, l'annotation ou le système automatique commence par une segmentation intégrale de chaque document en EDU qui sont ensuite attachées pour former une structure dans laquelle tous les segments sont liés les uns aux autres. Pour le PDTB, le processus est centré sur

les connecteurs et différents principes d'annotation font que la couverture n'est pas totale. On parle en général de *chunking discursif* (*discourse chunking*) (WEBBER et al., 2010) ou d'*analyse discursive de surface* (*shallow discourse parsing*, définition de la tâche partagée organisée lors de la conférence CoNLL 2015) pour la tâche consistant à reproduire les annotations du PDTB et d'*analyse discursive automatique* (*discourse parsing*) pour les autres corpus. Cette terminologie rend compte des liens que l'on peut établir entre l'analyse syntaxique des phrases et l'analyse discursive des documents. Nous adoptons donc désormais cette terminologie et décrivons dans les parties suivantes les analyseurs et les chunkers discursifs complets ainsi que les systèmes dédiés aux tâches intermédiaires, en réservant cependant une description plus détaillée des systèmes d'identification des relations implicites pour la suite de ce chapitre. Nous présentons dans une seconde partie les tâches de TAL et les applications qui se sont appuyées sur une analyse discursive automatique complète ou partielle.

3.1.1 Analyseurs discursifs

Un analyseur discursif complet comporte un module de segmentation du document en EDU et un système permettant de construire une structure étiquetée couvrant tout le document. La tâche s'apparente à de l'analyse syntaxique qui part d'une segmentation en mots et fournit une structure couvrant la phrase. En conséquence, les systèmes mettent généralement en œuvre des techniques adaptées de l'analyse syntaxique et rencontrent des problèmes similaires : l'explosion combinatoire, qui correspond au fait que le nombre d'analyses possibles pour une phrase/un document augmente exponentiellement avec le nombre de mots/d'EDU, et la définition des critères de validité d'une structure — des étapes de post-traitement ont ainsi pu être mis en œuvre pour prendre en compte les contraintes comme l'adjacence en RST ou la frontière droite en SDRT. La représentation des données nécessite une adaptation des analyseurs développés en syntaxe, car les unités élémentaires sont des segments textuels et non des mots. De plus, l'analyse discursive est probablement plus influencée par les propagations d'erreurs de l'étape de segmentation. Notons que les études donnent rarement des scores pour la seule tâche d'identification de la relation, car les étapes d'attachement et d'étiquetage ne sont en général pas séparées, l'algorithme prédisant un attachement qui détermine la structure et l'étiquette (comprenant éventuellement la nucléarité). Pour des scores globaux d'accord inter-annotateurs sur le RST DT, on trouve généralement le chiffre de 98% pour l'étape de segmentation, de $88,70\%$ pour la construction de la structure globale sans relation ni nucléarité et de $77,72\%$ pour cette structure avec la nucléarité seulement. La tâche complète, avec annotation des étiquettes de relations, correspond à un accord de $65,75\%$. Pour ANNODIS, AFANTENOS et al. (2012a) rapportent un accord de 66% en F_1 pour la tâche d'attachement (la construction de la structure sans étiquette de relation) et un coefficient kappa de $0,4$ pour la tâche complète avec 17 relations et de $0,57$ en se limitant à la distinction entre relations coordonnantes et subordonnantes.

3.1.1.1 Segmentation

La segmentation se décompose en général en un découpage du document en phrases — considéré comme un problème plus ou moins résolu (PALMER et HEARST, 1997 ; GILLICK, 2009) — puis des phrases en EDU — un problème plus complexe étant donné la relative diversité de ces unités. L'approche statistique est de loin la plus fréquente, mais on peut citer le système par règles développé par MARCU (2000) sur le premier corpus construit dans le cadre de la RST (MARCU et al., 1999) qui obtient une exactitude assez haute, $90,3\%$, en se fondant sur les connecteurs et la ponctuation. Marcu avait par ailleurs précédemment construit sur ce même corpus un système statistique, dont les performances étaient largement supérieures (97%) (MARCU, 1999). Le système se fonde sur des informations lexico-syntaxiques et un algorithme de classification par arbre de

décision pour identifier l'une des cinq classes suivantes : un token (i.e. un mot ou un signe de ponctuation) est une frontière d'EDU, de phrase, commence une unité parenthétique (correspondant aux unités enchâssées) ou la finit, ou n'entre dans aucune de ces catégories (token interne).

Les études suivantes, sur le RST DT, ont modélisé la tâche comme un problème de classification binaire sur les tokens (i.e. un token constitue ou non une frontière d'EDU). SORICUT et MARCU (2003) proposent un système de segmentation intra-phrastique de ce type fondé sur un modèle génératif utilisant des informations lexico-syntaxiques et rapportent une F_1 de $84,7\%$. Ces performances sont améliorées d'environ 2% par SAGAE (2009) avec un perceptron moyenné et des informations d'une analyse en dépendances. Par ailleurs, FISHER et ROARK (2007) ont montré l'utilité des informations dérivées de l'arbre syntaxique pour enrichir les informations issues d'étiqueteurs morpho-syntaxiques et de chunkers utilisées dans (SPORLEDER et LAPATA, 2005). Cependant, JOTY et al. (2012) ont proposé un modèle de segmentation intra-phrastique qui utilise moins de traits mais une méthode d'agrégation pour gérer le problème de déséquilibre des données, ce modèle parvenant à des performances similaires, de l'ordre de $90,5\%$. Pour la segmentation intégrale du document, SAGAE (2009) rapportent un score de F_1 de $92,9\%$ et HERNAULT et al. (2010) de $95,0\%$ avec une analyse syntaxique manuelle et d'environ 94% avec une analyse automatique en utilisant le jeu de traits défini dans (SORICUT et MARCU, 2003) mais en augmentant le contexte (i.e. la taille de la fenêtre autour du token considéré) et un algorithme SVM. Les performances sur cette tâche sont donc élevées mais n'atteignent pas encore les performances humaines (98%).

Notons que le traitement des unités enchâssées n'est pas clair dans les études sur le RST DT : il semble que le problème ait été ignoré dans toutes les études fondées sur une classification binaire de la tâche (STEDE, 2011), pourtant les analyseurs incluent clairement la relation *Same-unit* destinée à rattacher des segments discontinus d'une EDU. Comme nous l'avons dit dans le chapitre précédent, les unités dites enchâssées ne le sont pas forcément, donc il est possible que ces études n'aient considéré que les cas où cette relation liait des unités consécutives. Sur ANNODIS, AFANTENOS et al. (2010) proposent un système de segmentation qui doit gérer cette difficulté puisque le corpus comporte environ 10% d'EDU enchâssées. Au lieu de construire un classifieur binaire, les auteurs proposent un modèle à quatre classes correspondant au fait que le token commence un segment, finit un segment, est un segment à lui tout seul ou est un token interne. Le classifieur repose sur des informations lexico-syntaxiques et des indicateurs de présence de connecteurs discursifs. Les auteurs ajoutent une étape de post-traitement pour s'assurer de la cohérence de la segmentation. Au final, le système correspond à des performances de l'ordre de $73,3\%$ en F_1, montrant que la prise en compte des unités enchâssées rend la tâche plus difficile.

3.1.1.2 Construire l'arbre discursif, cadre de la RST

Dans le cadre du RST DT, l'étiquetage en relations peut se faire en utilisant l'intégralité du jeu de relations du corpus — donc 78 relations, inventaire étendu à 110 pour prendre en compte la nucléarité [2] —, ou des classes plus générales — les 16 classes définies dans (CARLSON et al., 2001), inventaire étendu à 18 pour inclure deux relations concernant l'organisation textuelle (*Textual-organization* et *Same-unit*) et à 41 pour prendre en compte la nucléarité. Comme pour la segmentation, l'attachement et l'étiquetage ont généralement été modélisés dans le cadre de systèmes d'apprentissage automatique. On peut cependant citer les systèmes par règles de MARCU (2000), développé sur le corpus de MARCU et al. (1999), et de LE THANH et al. (2004), sur le RST DT. Le premier s'appuie sur les connecteurs discursifs pour l'identification des liens — indices cependant non suffisants —, et sur une formalisation des structures discursives spécifiant des contraintes (principe de nucléarité, contrainte d'adjacence, de binarité des arbres, etc. . .) formulées

2. La relation *Evaluation* définie dans le RST DT correspond ainsi à une version multi-nucléaire et deux versions mononucléaires (CARLSON et MARCU, 2001), on a donc dans les systèmes trois relations *Evaluation*.

en logique du premier ordre — formalisme qui n'est cependant pas suffisant pour gérer l'explosion combinatoire. LE THANH et al. (2004) enrichissent ce système par la mise en œuvre d'une approche multi-niveaux — les arbres discursifs sont construits au niveau des phrases, puis des paragraphes et enfin du texte entier en respectant la contrainte d'adjacence —, approche qui ne permet cependant pas non plus de gérer l'ensemble de la combinatoire des analyses et n'est donc pas applicable à des textes longs.

Dans le cadre d'un système statistique, on peut distinguer trois grandes approches. MARCU (1999) a proposé d'utiliser, sur le corpus décrit dans (MARCU et al., 1999), un algorithme par transition (*shift-reduce*) : chaque EDU du document est considérée l'une après l'autre et l'algorithme décide soit de transférer cette EDU dans une pile (opération *shift*) soit de fusionner les éléments de la pile (EDU ou sous-arbres) en un nouveau sous-arbre (opération *reduce*). Plus précisément dans ce cas, le système comporte, en plus de l'opération de *shift*, un certain nombre d'opérations de type *reduce* liées à l'étiquette de relation et aux informations de nucléarité [3]. L'apprentissage des séquences d'actions se fait à partir d'un algorithme par arbre de décision en utilisant un ensemble assez large de traits : informations lexicales, syntaxiques, structurelles (comme le nombre de sous-arbres actuellement dans la pile) mais aussi de similarité sémantique (comme la mesure cosinus entre les segments en cours de traitement ou des mesures de similarité utilisant Wordnet reflétant la présence de synonymes, d'antonymes, etc...). SAGAE (2009) reprennent l'idée d'utiliser un algorithme de type shift-reduce mais n'intègrent que des informations lexico-syntaxiques extraites en se fondant sur l'analyse en dépendances et utilisent un perceptron moyenné pour l'apprentissage. Cette approche leur permet d'obtenir, avec une analyse syntaxique et une segmentation automatique, un score de F_1 de $52,9\%$ pour 18 relations [4] pour les cas intra-phrastiques et de $44,5\%$ pour l'analyse complète des documents.

Le système de SORICUT et MARCU (2003) (nommé SPADE), limité au cadre intra-phrastique, correspond à une autre modélisation du problème. Il se fonde sur un modèle d'analyse (*parsing model*) qui assigne une probabilité à chaque arbre candidat, et un analyseur (généralement appelé *discourse parser*) qui cherche le meilleur arbre parmi les candidats. Le modèle d'analyse est un modèle génératif qui se fonde sur des informations lexico-syntaxiques et structurelles, l'analyseur utilise une approche de programmation dynamique pour gérer l'explosion combinatoire. Cette étude a montré l'importance des informations tirées d'une analyse syntaxique en constituants. Avec une analyse syntaxique et une segmentation automatiques, les auteurs obtiennent des scores de F_1 de $70,5\%$ pour la structure sans relation, de $49,0\%$ en se restreignant à 18 relations (donc inférieur aux $52,9\%$ rapportés par SAGAE (2009)), et de $45,6\%$ avec 110 relations. Lorsque ces prétraitements viennent d'une annotation manuelle, les scores sont bien plus élevés (respectivement $96,2\%$, $75,5\%$ et $70,3\%$).

Une troisième approche se fonde sur des classifieurs en cascade, l'un gérant l'attachement l'autre l'étiquetage en relation. C'est la méthode mise en place par HERNAULT et al. (2010) (système HILDA) : la liste contient initialement toutes les EDU ; des scores d'attachement sont calculés pour tous les éléments consécutifs de la liste et la paire ayant reçu le plus haut score est étiquetée avec une relation et fusionnée en un sous-arbre ; le processus se répète jusqu'à ce que la liste ne contienne plus qu'un seul arbre couvrant l'ensemble du texte. HERNAULT et al. (2010) utilisent des classifieurs de type SVM et un jeu de traits inspirés de celui de SORICUT et MARCU (2003).

3. MARCU (1999) définit ainsi 102 opérations de type *reduce*, pour 17 groupes de relation étendus pour prendre en compte les 6 différentes configurations liées à la nucléarité (rappelons que cette étude n'est pas menée sur le RST DT).

4. Nous ne sommes pas sûre de comprendre si la mention de ce chiffre dans cette étude comme dans (SORICUT et MARCU, 2003) signifie que la nucléarité n'est pas prise en compte ou s'il s'agit juste d'une simplification dans la présentation des données. SORICUT et MARCU (2003) présentent par ailleurs également des scores pour 110 relations, donc incluant la nucléarité. Il est donc probable que ces deux études, quand elles se limitent aux 18 groupements, considèrent en fait les 41 relations.

Les auteurs rapportent des scores de F_1 de $47,3\%$ pour la structure complète avec l'ensemble de 41 relations et une segmentation automatique, et de $54,8\%$ avec une segmentation manuelle. Les auteurs rapportent par ailleurs un score d'exactitude de $66,8\%$ (macro-F_1 de $47,7\%$) pour la tâche d'identification des relations comprenant l'identification de la nucléarité. FENG et HIRST (2012) montrent que l'introduction dans ce système de nouveaux traits et des filtres sur ces traits proposés dans une étude sur l'identification des relations implicites dans le PDTB (LIN et al., 2009) permet d'améliorer les étapes d'attachement et d'identification des relations. Ils utilisent notamment les traits contextuels qui permettent de représenter les relations autour des segments courants et montrent ainsi les dépendances qui existent entre les sous-structures au sein d'un document. Le système complet fondé sur ces nouveaux traits (FENG et HIRST, 2014) correspond à une amélioration d'environ 5% par rapport à (HERNAULT et al., 2010).

Les systèmes suivants ont cherché à tirer profit des conclusions précédentes, tant au point de vue algorithmique que de modélisation des données. De plus, ils ajoutent généralement une distinction entre les niveaux inter- et intra-phrastiques d'analyse. Il a en effet été noté qu'il existe une forte corrélation entre frontières de phrase et frontières de discours, du moins dans le RST DT : dans ce corpus, une phrase correspond à une structure discursive complète bien formée dans 95% des cas (SORICUT et MARCU, 2003) [5]. JOTY et al. (2013) et JOTY et al. (2015) proposent ainsi deux modèles d'analyse (inter- et intra-phrastiques) entraînés séparément — afin de refléter les spécificités de chacun des niveaux d'analyse, par exemple la distribution des relations —, de type CRF — donc prenant en compte des dépendances séquentielles pour refléter les interactions entre les sous-structures —, et utilisant des traits surfaciques, lexico-syntaxiques et contextuels. L'analyseur utilise une approche par programmation dynamique (SORICUT et MARCU, 2003). Les auteurs proposent deux approches afin de combiner les analyses inter- et intra-phrastiques : la première suppose qu'une phrase correspond toujours à un arbre bien formé qui est donc fourni en entrée au système d'analyse inter-phrastique, la seconde consiste à fournir à l'analyseur inter-phrastique à la fois l'arbre pour la phrase courante et ceux correspondant aux phrases immédiatement précédentes et suivantes, ce qui a en fait tendance à introduire du bruit dans l'étiquetage des relations. Les auteurs rapportent au mieux une F_1 de $55,87\%$ (segmentation manuelle), améliorant d'environ 1% le système de HERNAULT et al. (2010). Les auteurs soulignent que l'étiquetage en relations correspond à des difficultés spécifiques comme le déséquilibre des classes et la confusion entre des relations similaires. Ces auteurs rapportent également des scores sur un corpus de manuels d'instructions (SUBBA et DI EUGENIO, 2009) de taille encore plus restreinte (176 documents) : le score obtenu, $44,20\%$, montre le caractère crucial de futurs travaux pour l'adaptation des parsers actuels à de nouveaux domaines.

Selon FENG et HIRST (2014), les modèles de type CRF et l'analyseur utilisé dans (JOTY et al., 2013 ; JOTY et al., 2015) posent cependant des problèmes d'efficacité. Ils proposent donc de l'améliorer en utilisant une approche similaire à celle proposée dans (HERNAULT et al., 2010), tout en conservant le principe d'utiliser des CRF (à la place des SVM) aux niveaux inter- et intra-phrastiques. Le système améliore d'environ $1,5\%$ les performances présentées dans (JOTY et al., 2013). L'ajout d'une étape de post-traitement à chaque niveau, prenant en compte la profondeur d'un nœud dans l'arbre et permettant de réévaluer les décisions d'attachement, amène une amélioration supplémentaire avec un score d'exactitude final de $58,2\%$ (segmentation manuelle). L'analyse d'erreur montre une difficulté à identifier certaines relations de sémantique faible comme *Textual-organization* ou *Topic-change*. L'idée d'un post-traitement des structures produites par un premier analyseur discursif a aussi été explorée dans (JOTY et MOSCHITTI, 2014) avec l'introduction d'un modèle permettant de réviser les structures prédites en tenant compte de la structure globale dont

5. Les autres cas correspondent à une configuration où une unité est attachée à la structure d'une autre phrase avant d'être liée aux unités de la phrase à laquelle elle appartient, cas qui semblent présenter un caractère assez systématique (VLIET et REDEKER, 2011).

les propriétés sont représentées à l'aide de noyaux d'arbre. Ceci permet notamment de retrouver des dépendances à longue distance. L'ajout de cette étape permet d'améliorer le système proposé dans (JOTY et al., 2013) avec un score final de $57,3\%$ de F_1 (segmentation manuelle), cependant inférieur à celui présenté dans (FENG et HIRST, 2014).

Le meilleur système actuel (JI et EISENSTEIN, 2014b) doit essentiellement ses performances à un enrichissement de la modélisation des données, qui permet de mieux refléter les distinctions fondamentalement sémantiques entre les relations discursives. Les auteurs reprennent l'approche utilisant un algorithme par transition (MARCU, 2000 ; SAGAE, 2009), avec cependant un algorithme à large marge pour l'apprentissage, et proposent d'apprendre, en plus des séquences d'actions, une transformation de la représentation surfacique des données (i.e. les mots constituant les unités). La prédiction s'effectue en maximisant le produit scalaire du vecteur de paramètres avec un vecteur obtenu en transformant le vecteur des EDU — résultat de la concaténation des vecteurs représentant l'unité courante et les deux unités au sommet de la pile — à l'aide d'une fonction (dite *de représentation*) paramétrée par une matrice. La transformation est une projection linéaire de la représentation surfacique dans un espace latent de taille réduite. Trois variantes de la matrice sont considérées, selon que l'on n'impose aucune contrainte sur sa forme, que l'on force une même transformation mais séparée des EDU ou que l'on s'intéresse à la représentation des différences entre les EDU adjacentes, capturant ainsi leur similarité. L'utilisation de cette transformation de la représentation, combinée à d'autres traits lexico-syntaxiques, permet d'améliorer les performances des systèmes précédents, le système complet obtenant un score final de F_1 de $61,63\%$ (segmentation manuelle). Cette amélioration reflète le caractère crucial de la représentation des données et révèle le potentiel de la représentation fondée sur les mots. L'idée de transformer les unités en une représentation plus abstraite est reprise par LI et al. (2014a) qui utilisent un modèle de réseau de neurones récursif (RNN) opérant sur des entrées structurées et permettant de produire une représentation vectorielle pour chaque unité discursive. Les auteurs reprennent le principe de deux classifieurs pour l'attachement et l'étiquetage proposé dans (HERNAULT et al., 2010). Ils obtiennent des performances au niveau de l'état de l'art, inférieures cependant à celles obtenues dans (JI et EISENSTEIN, 2014b).

Le développement d'analyseurs discursifs s'est intensifié ces dernières années, reprenant les méthodes ayant montré leur efficacité en analyse syntaxique ou pour d'autres tâches plus sémantiques. Le meilleur système (JI et EISENSTEIN, 2014b) correspond à une performance globale de l'ordre de $61,63\%$ en F_1, donc relativement proche des performances humaines estimées à 66%. Il nous semble que l'un des enjeux majeurs de ces systèmes réside dans le manque de données, tant au niveau des documents qui constituent l'unité d'analyse, au nombre de 385 seulement dans le RST DT, que des instances de relations, étant donné le grand nombre de classes à considérer. Les futurs systèmes, et les systèmes développés pour d'autres langues pour lesquelles le manque de données est plus important encore, devraient donc mettre en place de nouvelles stratégies pour gérer ces difficultés.

3.1.1.3 Construire le graphe discursif, cadre de la SDRT

MULLER et al. (2012a) construisent un analyseur discursif sur le corpus ANNODIS dans le cadre de la SDRT, la structure finale est donc un graphe direct acyclique (DAG). Les auteurs utilisent des techniques développées pour l'analyse syntaxique en dépendance. Pour cela, ils transforment l'annotation en remplaçant les CDU par leur tête qui correspond à l'EDU la plus haute dans sa hiérarchie interne, le processus étant récursif. Deux modèles locaux sont construits pour les tâches d'attachement et d'étiquetage, les sorties de ces classifieurs sont ensuite envoyées à un système de décodage. Ces modèles utilisent des traits classiques, comme la position de chaque EDU, leur

longueur et des informations syntaxiques et temporelles ainsi que des informations concernant les connecteurs présents. Ils utilisent aussi des informations concernant la structure comme la distance entre les EDU, l'enchâssement de l'EDU source dans l'EDU cible et inversement ou l'enchâssement de l'EDU cible dans n'importe quelle autre EDU. Les structures sont ensuite construites en utilisant l'algorithme d'arbre couvrant maximal (MST) (McDonald et Pereira, 2006) ou un algorithme de recherche du plus court chemin parmi un ensemble de solutions (A^*) qui optimise un critère global et permet d'encoder des contraintes sur la structure comme la contrainte de la frontière droite. Les auteurs présentent aussi des résultats en utilisant une approche similaire à celle de Hernault et al. (2010). Ils montrent que l'utilisation des algorithmes d'analyse en dépendance permet de surpasser cette approche simple. Leur meilleur système correspond à un score de F_1 de $66,2\%$ pour la structure non étiquetée et $46,8\%$ pour la structure complète lorsque 4 grands groupes de relations sont considérés et $36,1\%$ pour l'ensemble des 18 relations. Il est clair ici que la taille restreinte du corpus (86 documents) a un impact important sur les résultats. Muller et al. (2012a) donnent également des résultats pour les sous-tâches d'attachement et d'identification des relations. Pour la première, ils rapportent un score de F_1 de $63,5\%$ et, pour la seconde, de $43,3\%$ en considérant les 18 relations du corpus et de $63,6\%$ en les groupant en 4 grandes classes. Les auteurs présentent leur système comme généralisable à d'autres cadres comme celui de la RST : la structure discursive d'un document dans le RST DT, similaire à une analyse syntaxique en constituance, peut en effet être traduite en un graphe de dépendance en considérant les segments récursivement selon leur valeur de nucléarité.

3.1.2 Chunkers discursifs

Le chunking discursif est une approximation de l'analyse discursive telle que fournie par les systèmes précédents car elle n'aboutit qu'à une analyse partielle des documents, elle ne produit pas une structure globale dans laquelle chaque segment est lié à un autre. Webber et al. (2010) en donnent ainsi la définition suivante, que nous traduisons : « Le terme de *chunking discursif* réfère à la reconnaissance dans un discours d'unités comme des *relations discursives* qui ne sont pas supposées fournir une couverture *totale* du texte ». Le terme *relation discursive* est ici à comprendre au sens d'une unité correspondant à deux segments de texte liés, ce lien pouvant être étiqueté par un sens (ou type de la relation). Webber et al. (2010) décrivent les connecteurs discursifs comme des indices d'une telle relation. Il est cependant difficile d'inclure les exemples non explicites dans ce cadre à moins de construire un système résolvant la tâche d'attachement, donc identifiant les connecteurs vides, rarement modélisée de façon complète sur le PDTB. Cette définition des chunkers discursifs est donc claire quand on considère les relations explicites, on identifie le connecteur, son sens et ses arguments. Pour les autres types de relations, l'identification est facilitée par l'annotation actuelle du PDTB qui ne prend en compte que des phrases adjacentes non liées. Si l'annotation venait à s'enrichir de relations implicites intra-phrastiques, des systèmes de segmentation et d'attachement du type de ceux construits pour le RST DT deviendraient nécessaires.

Reproduire les annotations du PDTB correspond à plusieurs tâches généralement étudiées séparément : l'identification des connecteurs, de leurs arguments et de la relation qu'ils déclenchent, et l'identification des relations non explicites. Le problème de la délimitation précise des arguments des relations non explicites n'a pas, à notre connaissance, été traité (rappelons que le principe de minimalité et le traitement de l'attribution font que ces arguments ne sont pas exactement deux phrases adjacentes). L'identification des informations d'attribution n'a pas non plus été complètement traitée, Lin et al. (2010) se limitant à la délimitation des segments attributifs. Nous présentons ici les systèmes consacrés aux exemples explicites ainsi que le système complet de (Lin et al., 2010) [6].

6. Ce rapport a été présenté plus tard sous la forme d'un article (Lin et al., 2014)

3.1.2.1 Identification des connecteurs

L'identification des connecteurs correspond la désambiguïsation en emploi des expressions incluses dans la liste pré-établie de connecteurs. Pour simplifier on parlera, par abus de langage, de désambiguïsation en emploi des connecteurs. Cette tâche est modélisée comme un problème de classification binaire : la forme est en emploi discursif (i.e. est un connecteur) ou non. Les principales informations fournies au modèle concernent le connecteur (potentiel), les mots apparaissant autour de lui et leur catégorie morpho-syntaxique, et des informations syntaxiques (par exemple, la présence d'un VP dans un voisin du connecteur ou le chemin entre lui et la racine de l'arbre). Aucun système de désambiguïsation en emploi ne prend en compte les arguments potentiels des formes considérées. PITLER et al. (2009) ont proposé le premier système de ce type, ils montrent qu'un modèle ne prenant comme trait que la chaîne de caractères correspondant au connecteur conduit déjà à des performances de l'ordre de $85,86\%$ en exactitude ($75,23\%$ en F_1). L'ajout des traits morpho-syntaxiques et de combinaisons entre ces traits permet d'améliorer ces scores jusqu'à $96,26\%$ en exactitude ($94,19\%$ en F_1). LIN et al. (2010) améliorent ces scores d'environ 1% en ajoutant des traits correspondant à des combinaisons des chaînes de caractères et des catégories morpho-syntaxiques du connecteur et des mots l'entourant. Enfin, IBN FAIZ et MERCER (2013) obtiennent une très légère amélioration ($97,34\%$ en exactitude, $96,22\%$ en F_1) en considérant notamment les étiquettes de constituants dominant les voisins plutôt que leurs catégories morpho-syntaxiques, et en divisant le chemin dans l'arbre en plusieurs traits.

Ces bonnes performances par rapport à la simplicité des modèles montrent que les connecteurs sont assez peu ambigus, et nous verrons que c'est aussi le cas pour les relations qu'ils déclenchent. Cette conclusion doit cependant être nuancée. Les performances chutent lorsque l'on ne dispose plus d'une analyse syntaxique gold, LIN et al. (2014) rapportent ainsi une baisse d'environ $1,3\%$ en exactitude et 2% en F_1. De plus, JOHANNSEN et SØGAARD (2013) notent l'importance de prendre en compte les performances par connecteur : les systèmes existants ont surtout de bonnes performances pour les connecteurs très fréquents. Cette situation engendre deux problèmes. D'abord, les auteurs notent que 70% des articles constituant le corpus contiennent au moins un connecteur qui n'est pas parmi les 10 connecteurs les plus fréquents (75% des données). Or, se tromper sur un connecteur c'est potentiellement perdre ou ajouter un lien ce qui engendre donc une erreur sur la structure entière. Comme cette identification constitue la toute première étape d'un chunker discursif, une erreur à ce niveau va se propager dans l'ensemble de la chaîne. Ensuite, il faut noter que la distribution des connecteurs est très probablement différente selon les domaines et que l'on ne peut donc pas simplement ignorer les connecteurs peu fréquents dans le *Wall Street Journal* car ils pourraient être fréquents dans d'autres textes. JOHANNSEN et SØGAARD (2013) proposent un modèle plus simple, notamment sans informations syntaxiques, et s'évaluent en utilisant une analyse morpho-syntaxique automatique. Les performances rapportées en termes de F_1 par connecteur montrent que la tâche est loin d'être résolue pour les connecteurs moins fréquents. Par exemple, le cinquantième connecteur par ordre de fréquence, *ultimately*, correspond à une F_1 de l'ordre de 30% seulement.

3.1.2.2 Identification des arguments des connecteurs

Les premières approches se sont limitées à l'identification des têtes des arguments : le connecteur est vu comme un prédicat prenant ces têtes comme arguments. WELLNER et PUSTEJOVSKY (2007) modélisent cette tâche comme un problème de classification binaire, avec un modèle par argument. L'apprentissage se fonde sur des informations surfaciques et syntaxiques, provenant d'analyse en constituants et en dépendances. Ils obtiennent au mieux un score d'exactitude de $67,9\%$ pour l'argument 1 et $90,6\%$ pour l'argument 2. Les auteurs proposent ensuite de combiner les probabilités obtenues pour chaque candidat et chaque argument (par multiplication), afin

de dégager la paire d'arguments la plus probable. Ce système conduit à une amélioration des performances pour le premier argument ($69,8$%). Le premier argument est plus difficile à identifier que le second qui contient le connecteur et qui lui est donc syntaxiquement lié. Notamment, le problème le plus complexe est l'identification du premier argument des adverbiaux. L'utilisation de modèles spécifiques, correspondant à la catégorie du connecteur entre coordonnant, subordonnant et adverbial, combinés linéairement a permis une amélioration des performances notamment pour le premier argument avec $80,0$% en exactitude (ELWELL et BALDRIDGE, 2008). En plus de sa catégorie, la position du connecteur a aussi un impact. PRASAD et al. (2010) présentent des résultats pour les connecteurs inter-phrastiques en se limitant à la localisation de la phrase contenant le premier argument. Leur stratégie consiste à filtrer l'ensemble des phrases candidates en définissant des zones dites « opaques » bloquant la portée du connecteur considéré et à utiliser des informations de coréférence. Ils améliorent ainsi de 3% un système de référence.

Les études suivantes ont cherché à localiser (localisation par rapport au connecteur) mais aussi à identifier l'empan des arguments des connecteurs (étape dite d'extraction), tâche particulièrement difficile dans le PDTB notamment à cause du principe de minimalité. Notons cependant que l'accord inter-annotateur est assez haut, $90,2$% en correspondance exacte et $94,5$% en correspondance partielle (PRASAD et al., 2008a). Les systèmes séparent en général le cas de l'argument 2, l'argument auquel le connecteur appartient, et celui de l'argument 1. Ainsi, le système de localisation de LIN et al. (2014) est concentré sur l'argument 1, et correspond à un modèle de classification binaire (l'argument est dans la même phrase que le connecteur ou dans une phrase précédente) utilisant des informations lexico-syntaxiques. LIN et al. (2014) rapportent $97,94$% de F_1 avec un connecteur manuellement identifié et $94,35$% avec un connecteur prédit. L'extraction correspond à l'identification de l'argument auquel appartient un nœud dans l'arbre syntaxique et se fonde aussi sur un classifieur binaire et des informations sur le connecteur et la structure de l'arbre syntaxique [7]. En correspondance partielle et en utilisant les connecteurs annotés, les scores sont assez hauts : $86,67$% de F_1 pour l'argument 1, $99,13$% pour l'argument 2 et $86,24$% de score général. Avec propagation d'erreurs (connecteurs prédits et analyse syntaxique automatique), le score global est de $80,96$% de F_1. L'évaluation en correspondance exacte, même sans propagation d'erreurs, donne des scores bien inférieurs : le score global n'est que de $53,85$% avec une baisse de près de 17% pour l'argument 2 et de plus de 25% pour l'argument 1. Le score final pour un système entièrement prédit avec correspondance exacte est de $40,37$% de F_1.

GHOSH et al. (2011) restent sur une identification séparée des deux arguments mais modélisent la tâche comme un étiquetage séquentiel au niveau des tokens. Ils utilisent notamment le sens du connecteur comme trait, en plus d'informations lexico-syntaxiques, et les étiquettes des constituants de l'argument 2 pour l'identification de l'argument 1, introduisant une information d'ordre structurel. Dans une étude suivante (GHOSH et al., 2012), ils ajoutent notamment une phase de réordonancement des résultats qui permet la prise en compte d'informations globales ou de contraintes (chaque connecteur doit avoir deux arguments, l'argument 1 est généralement avant l'argument 2 etc...). Malgré ces enrichissements, ils ne parviennent pas à dépasser les scores de LIN et al. (2014). Par contre, KONG et al. (2014) obtiennent des améliorations en utilisant une approche fondée sur les constituants au lieu des mots. Ils unifient le traitement des cas inter- et intra-phrastiques en considérant la phrase précédent l'argument 1 comme un constituant spécial et utilisent un mécanisme d'inférence jointe pour tenir compte d'informations plus globales (des contraintes comme le non chevauchement des arguments ou un maximum de deux constituants discontinus pour un argument). Ils combinent de plus deux analyseurs syntaxiques afin de gérer les potentielles erreurs à ce niveau. Ils améliorent les résultats de LIN et al. (2014) en correspondance exacte d'environ 5% de F_1 pour le score général, et d'environ 6% et $1,5$% pour l'argument 1 et 2.

7. Ce modèle n'est appliqué que dans le cas où l'argument 1 est dans la même phrase que le connecteur, sinon l'argument 1 correspond à la phrase immédiatement précédente (système de référence par majorité qui conduit à $76,9$% de F_1).

3.1.2.3 Identification des sens des connecteurs

Comme pour la désambiguïsation en emploi, la désambiguïsation en relation des connecteurs est généralement effectuée sans faire appel à leurs arguments. Ici, la situation semble encore plus idéale que précédemment puisqu'un système ne considérant que la chaîne de caractères constituant le connecteur obtient un score d'exactitude de $93,67\%$ (PITLER et al., 2009) au niveau 1 de relation (quatre classes), ce score n'étant que très légèrement amélioré par l'ajout de traits lexico-syntaxiques, au mieux $94,15\%$ dans (PITLER et al., 2009). Notons qu'ici les auteurs utilisent un classifieur multiclasse (de type Naïf Bayes), contrairement à la configuration la plus répandue pour les implicites au niveau 1 (un classifieur binaire par classe). Ces auteurs présentaient, dans (PITLER et al., 2008), des scores en binaire en utilisant seulement le connecteur comme trait et des classifieurs par arbre de décision, ils obtenaient des scores d'exactitude de $97,23\%$ pour *Comparison*, $93,99\%$ pour *Contingency*, $95,4\%$ pour *Temporal* et $97,61\%$ pour *Expansion*. Les scores rapportés par WANG et al. (2010), en multiclasse (classifieur SVM) sont beaucoup plus bas, leur meilleur système, pourtant fondé sur une représentation par noyau des informations syntaxiques, des traits de coréférence et des informations temporelles, ne correspond qu'à une exactitude de $74,2\%$ pour les relations explicites. Il se peut que cette différence vienne de l'inclusion du problème d'attachement dans le système de classification (l'ajout d'une étiquette indiquant l'absence de relation).

Pour le niveau 2 de relations (16 relations), LIN et al. (2010) rapportent, également avec un classifieur multiclasse mais de type régression logistique, des scores de précision et de F_1 de $86,77\%$ en n'utilisant que le connecteur, sa catégorie morpho-syntaxique et le mot précédent, performance assez largement dégradée (environ 6%) avec une propagation des erreurs des étapes précédentes (analyse syntaxique et identification des connecteurs). VERSLEY (2011) présente une étude aux niveaux 1, 2 et 3 en utilisant un principe de classification hiérarchique et des informations (verbe, modalité, négation. . .) extraites des arguments identifiés à partir d'heuristiques. Ces enrichissements n'apportent cependant pas d'importantes améliorations. Pour le niveau 3, VERSLEY (2011) rapporte au mieux 80% de F_1. Par ailleurs, cette étude montre que les scores globaux occultent le fait que certaines relations, les moins fréquentes, sont très mal identifiées comme *Concession* ou *List*.

Notons que ces scores sont largement supérieurs à ceux obtenus pour les implicites — de l'ordre de 60% d'exactitude au niveau 1 et de 40% au niveau 2 — reflétant l'importance de l'indice constitué par les connecteurs. Ces résultats montrent aussi que les connecteurs sont peu ambigus en termes de la relation qu'ils déclenchent. Pour le niveau 1, PITLER et al. (2008) notent ainsi que les connecteurs apparaissent avec leur sens dominant dans $93,43\%$ des cas pour *Comparison*, $94,72\%$ pour *Contingency*, $97,63\%$ pour *Expansion* et $84,10\%$ pour *Temporal*, cette dernière étant donc la plus difficile à identifier. L'accord inter-annotateur fourni ne sépare pas relations implicites, explicites et AltLex, il correspond à 94% pour le niveau 1, 84% pour le niveau 2 et 80% pour le niveau 3 (PRASAD et al., 2008a), performances qui sont donc atteintes pour les explicites ce qui n'est pas le cas pour les implicites.

3.1.2.4 Chuncker discursif complet

Il n'existe à l'heure actuelle qu'un seul chuncker discursif complet construit sur le PDTB (LIN et al., 2010 ; LIN et al., 2014) [8]. Ce système est conçu comme un ensemble de modules dont chacun fournit ses sorties au suivant souvent sans rétro-actions, les informations d'arguments ne sont ainsi pas utilisées par le module identifiant les relations explicites. L'identification des relations non

8. La tâche partagée évoquée va cependant bientôt introduire de nouveaux systèmes, systèmes qui ne seront cependant pas directement comparables au système de LIN et al. (2014) car les données sont différentes : l'ensemble d'évaluation n'est pas le même non plus que le jeu de relations. La tâche n'inclut pas non plus l'identification des segments attributifs.

explicites utilise cependant des traits reflétant des informations sur les relations explicites et leurs arguments.

Le premier module correspond à la désambiguïsation en emploi des connecteurs. Les deux modules suivants permettent de localiser les arguments et d'en identifier l'empan. Viennent ensuite deux modules servant à identifier les relations explicites et les relations non explicites, ces dernières englobant relations implicites, AltLex, EntRel et NoRel. Cette identification se fait au niveau 2 de la hiérarchie des relations du PDTB. Notons que, pour les implicites et les AltLex, seules onze relations sont prises en compte sur les seize possibles, les cinq exclues correspondant à trop peu d'exemples (LIN et al., 2009). Les auteurs ne détaillent pas l'identification des arguments pour les relations non explicites et ne donnent pas de scores, même si, comme on l'a vu, ces arguments ne correspondent pas toujours exactement aux phrases adjacentes non liées par un connecteur. On peut éventuellement avoir des morceaux marqués comme supplémentaires et/ou des ensembles de plusieurs phrases. Enfin, le dernier module sert à délimiter l'empan des segments attributifs pour les explicites, les implicites et les AltLex.

Comme nous avons pu l'évoquer en présentant les résultats dans les sections précédentes, ces auteurs évaluent chaque module avec et sans propagation d'erreurs des étapes précédentes, et avec une analyse syntaxique automatique ou manuelle. Le seul module que nous n'avons pas présenté est celui correspondant à l'identification des relations non explicites, modélisé comme un système de classification à 13 classes : 11 relations implicites/AltLex de niveau 2, EntRel et NoRel. Ils utilisent un classifieur par régression logistique et le jeu de traits est le même que dans leur étude consacrée aux implicites (PITLER et al., 2009), ils contient des informations lexicales, syntaxiques et contextuelles (les relations autour de celle conidérée). Les auteurs rapportent $39,63\%$ de F_1 avec une analyse syntaxique manuelle et sans propagation d'erreurs, et $25,64\%$ avec un système entièrement prédit.

Le système complet, avec analyse syntaxique manuelle, correspond à un score de F_1 de $46,80\%$ pour une correspondance partielle au niveau des arguments et de $33,00\%$ pour une correspondance exacte. L'utilisation de l'analyse syntaxique automatique conduit à une baisse d'environ 8% en correspondance partielle et de près de 13% en correspondance exacte. Notons que parmi tous les modules, celui correspondant à l'identification des relations non explicites donne les scores les plus bas, quelle que soit la configuration. Il est suivi par les modules qui déterminent l'empan des arguments des connecteurs ou des segments attributifs en correspondance exacte, tâche difficile en regard des conventions d'annotation du PDTB. L'intégration du système de KONG et al. (2014) pour l'identification des arguments conduit à une amélioration globale d'environ $1,8\%$ de F_1 avec une analyse syntaxique manuelle et de presque 7% avec une analyse prédite.

3.1.3 Applications

Dans cette section, nous présentons brièvement les différentes tâches de TAL et applications qui ont pu s'appuyer sur une analyse discursive complète ou partielle. Une bonne description des applications peut être trouvée dans (WEBBER et al., 2010 ; TABOADA et MANN, 2006a). Notons que le principe d'une analyse discursive des documents a été utilisé pour la génération automatique de textes. TABOADA et MANN (2006a) font référence à de nombreux projets qui ont fait usage des relations discursives pour guider le processus afin notamment d'assurer la cohérence du texte en sortie et de sélectionner les bons marqueurs. Par ailleurs, un formalisme, nommé G-TAG (DANLOS, 2000), s'inscrivant dans le cadre des TAG, a été développé pour la génération de texte, formalisme pour lequel le niveau inter-phrastique se fonde sur les connecteurs d'une manière similaire à ce qui est fait dans D-LTAG ou D-STAG.

La structure discursive peut fournir différents types d'informations pertinentes pour une variété d'applications. La définition d'un principe d'organisation hiérarchique entre les segments, reposant sur la notion de nucléarité en RST, permet d'identifier des segments plus importants ce qui peut aider des systèmes de résumé automatique (DAUMÉ III et MARCU, 2009 ; THIONE et al., 2004 ; SPORLEDER et LAPATA, 2005). C'est par ailleurs la tâche qui a motivé le développement du premier analyseur discursif (MARCU, 1997a ; MARCU, 1997b). Il a aussi été montré que la relation discursive pouvait être un indice supplémentaire pour identifier les parties importantes d'un texte (LOUIS et al., 2010a). Ensuite, des systèmes d'aide à la correction de rédaction se sont appuyés sur la notion de cohérence d'un document (HIGGINS et al., 2004), notion qui peut être évaluée par une analyse discursive (LIN et al., 2011), sur la structure discursive complète (BURSTEIN et al., 2003) ou sur certaines relations seulement (BURSTEIN et al., 1998). Les connecteurs discursifs ont été utilisés comme des indices pour déterminer la polarité d'une proposition puisqu'ils peuvent l'inverser (POLANYI et ZAENEN, 2006). L'analyse de sentiment a aussi cherché à profiter de la structure hiérarchique du discours, d'abord sans grand succès (VOLL et TABOADA, 2007). BHATIA et al. (2015) ont cependant montré récemment que des améliorations pouvaient être obtenues en se fondant sur les derniers analyseurs discursifs développés sur le RST DT. CHAI et JIN (2004) discutent des possibilités d'utilisation de l'analyse discursive pour des systèmes de question-réponse qui ont par exemple été explorées dans le cas des questions en *pourquoi* (VERBERNE, 2007). Enfin, d'autres systèmes de TAL ont mis en œuvre des stratégies reposant sur des informations discursives comme la détection de paraphrase (HIONG et al., 2012) ou la résolution de la coréférence (CRISTEA et al., 1999). Il est clair que les performances des analyseurs discursifs actuels limitent encore le développement d'applications fondées sur ce niveau d'analyse et que leur amélioration permettra d'accroître son apport et d'enrichir le panel des applications l'utilisant. Pour toutes ces applications, l'identification de la relation entre deux segments est cruciale, parce qu'elle détermine un lien hiérarchique ou parce que le sens qu'elle établit est informatif.

Finalement, notons que plusieurs travaux se sont intéressés aux connecteurs et à leur traduction. Les connecteurs ne sont pas faciles à traduire automatiquement du fait de leur ambiguïté, notamment quand un connecteur est ambigu dans une langue mais que son correspondant privilégié, selon les dictionnaires bilingues, ne l'est pas. MEYER (2011) propose de désambiguïser les connecteurs afin d'en fournir une meilleure traduction. De plus, dans les corpus parallèles, ils peuvent n'être traduits par aucune forme ou être insérés alors que le texte source n'en contenait pas, comme le montrent DANLOS et ROZE (2011) dans le cas de la traduction vers ou depuis l'anglais des connecteurs français *en effet* et *alors que*, ce qui fausse les scores des systèmes de traduction automatique. Notons également que, dans l'autre sens, la traduction automatique a été utilisée pour identifier les sens des connecteurs et enrichir un lexique de connecteurs (CARTONI et al., 2013 ; VERSLEY, 2010). L'existence des mêmes relations de discours dans toutes les langues, du moins pour les concepts les plus généraux, permet d'envisager l'utilisation de ce type de méthode pour construire des systèmes d'analyse discursive pour des langues peu dotées.

3.2 Identification automatique des relations implicites : importance, complexité et difficultés

Le développement de systèmes d'analyse discursive est donc crucial pour de nombreuses applications. L'étiquetage en relation, en plus d'être l'une des raisons des basses performances des systèmes complets, est particulièrement important pour ces applications. Dans cette section, nous montrons qu'il est nécessaire d'améliorer les performances pour l'identification des relations implicites car elles représentent une part importante des données (section 3.2.1). La tâche d'identification des relations discursives implicites est réputée complexe ce qui se reflète dans les performances.

La principale difficulté réside dans la nature et la diversité des indices qui doivent être pris en compte (section 3.2.2). De plus, les données manuellement annotées que nous utilisons présentent certaines caractéristiques qui posent des problèmes connus dans le cadre de méthodes d'apprentissage automatique : après avoir introduit certaines notions concernant l'apprentissage automatique (section 3.2.3), nous détaillons ces problèmes par ailleurs en partie liés aux indices utilisés (section 3.2.4).

3.2.1 Importance de l'identification des relations implicites

Dans les sections précédentes, nous avons présenté les systèmes d'analyse discursive complets ainsi que les différentes sous-tâches concernant la segmentation et les relations explicites. Nous avons vu que ces dernières correspondaient à des performances hautes alors que les systèmes complets souffrent d'une mauvaise identification des relations qui entraînent des performances générales relativement basses. Ceci reflète le caractère crucial de l'amélioration de l'identification des relations implicites dans le cadre d'un système complet.

On pourrait penser que s'il existe dans la langue des marques pas ou peu ambiguës permettant d'indiquer un lien entre les segments au sein d'un texte alors un locuteur aura tendance à les mettre en œuvre fréquemment afin d'assurer une communication efficace. Cependant, les relations implicites sont très répandues dans les textes. Dans le PDTB, sur les $40\,600$ exemples annotés, $45,5\%$ sont explicites, $39,5\%$ sont implicites. Le reste est constitué de $12,8\%$ d'EntRel, $1,5\%$ d'AltLex et $0,6\%$ de NoRel. Si on ne conserve que les implicites et les explicites, on obtient une proportion de $53,5\%$ d'explicites et donc $46,5\%$ d'implicites. Ces chiffres montrent qu'un système ne gérant que les explicites n'est capable d'identifier qu'environ une relation sur deux. Cette proportion est probablement dépendante de la langue et du domaine. Elle est aussi dépendante, bien sûr, du nombre de connecteurs considérés, et l'inventaire est relativement restreint dans le cas du PDTB (100 formes) par rapport, par exemple, au lexique français LexConn (plus de 300 formes). Notons cependant qu'en augmentant le nombre de connecteurs, en considérant des formes moins canoniques, on augmente probablement le taux d'ambiguïté en emploi et en relation. Notamment, si l'ambiguïté en relation augmente, les systèmes consacrés aux explicites devront s'enrichir de nouvelles informations du type de celles utilisées pour les implicites.

Malgré les spécificités du PDTB, et notamment le faible nombre de connecteurs considérés, la distribution à part égale des deux types de relations semble une estimation assez juste, les relations implicites pouvant même être dominantes. On retrouve des distributions similaires dans des genres différents et pour des langues différentes. Dans le British National Corpus, environ la moitié des phrases ne contiennent aucun connecteur (Sporleder et Lascarides, 2008). Toujours pour l'anglais, Taboada (2006) rapporte une proportion de 31% d'exemples explicités par une conjonction dans des dialogues et Subba et Di Eugenio (2009) une proportion de 43% pour un corpus de manuels d'utilisation.

Notre projection de LexConn sur le corpus français Annodis [9] contenant des articles journalistiques mais aussi de *Wikipedia*, nous donne une proportion d'exemples implicites comprise entre $47,4$ et 71% selon le jeu de relations choisies, c'est-à-dire selon que l'on prenne en compte ou pas les relations qui sont toujours implicites (selon l'inventaire de connecteur choisi) : le premier chiffre ne prend pas en compte des relations qui sont toujours implicites, ou du moins qui ne sont jamais marquées par un connecteur connu (*Attribution*, *Entity Elaboration* et *Frame*), tandis que le second chiffre prend en compte l'ensemble des relations. Pour l'italien, Soria et Ferrari

9. La projection du lexique correspond à l'identification automatique des connecteurs dans les données, identification bruitée car nous avons simplement considéré comme explicite une relation si l'un des arguments contenait une forme présente dans le lexique et pouvant déclencher la relation annotée.

(1998) mènent une expérience où ils demandent à 19 sujets de raconter à l'oral puis de mettre par écrit une histoire à partir d'une série d'images. Les auteurs comparent ensuite les versions et rapportent des proportions de relations explicitées de 48, 4% pour la version orale et de 37, 3% pour la version écrite, donc une proportion d'implicite dépassant les 50% pour les deux genres. Pour le corpus allemand annoté au niveau discursif (GASTEL et al., 2011), VERSLEY (2013) rapporte la présence de 65% d'exemples implicites. Dans un autre corpus allemand (STEDE, 2004) composé de commentaires donc de textes relativement courts et d'un genre spécfique, les auteurs rapportent que 35% des exemples sont marqués [10]. Une étude plus ancienne pour laquelle 37 textes en allemand avaient été examinés et annotés faisait un constat similaire avec 61, 2% de relations non marquées (SCHAUER et HAHN, 2001). Enfin, pour l'hindi, les auteurs du Hindi Discourse Relation Bank (OZA et al., 2009) rapportent une proportion de 31, 4% d'explicites et de 30, 7% d'implicites, le reste étant constitué de 6, 1% d'AltLex, de 23, 2% d'EntRel et de 8, 5% de NoRel.

Finalement, il faut noter que les relations ne sont pas toutes susceptibles d'être marquées avec la même fréquence. Certaines relations sont notoirement connues comme fréquemment marquées, par exemple *Concession* ou *Condition*, d'autres au contraire sont peu marquées ou ne correspondent pas à des marqueurs de type connecteur discursif comme *Background* ou *Frame*. TABOADA (2006) fait une étude sur six relations dans le RST DT dont la moitié correspond à des relations considérées comme souvent marquées (*Concession*, *Circumstance* et *Result*) et l'autre à des relations peu marquées (*Background*, *Elaboration* et *Summary*). Ils rapportent que les données correspondant à ces relations contiennent environ 31% d'exemples explicités par un connecteur mais les différences entre les relations sont importantes.

Les relations non marquées par un connecteur sont très fréquentes, elles doivent donc être prises en compte dans la construction d'un système d'identification des relations, système qui, nous l'avons vu est crucial pour la construction d'un analyseur discursif et qui peut aussi aider d'autres tâches de TAL ou applications. Cependant, nous avons évoqué les performances relativement basses des systèmes actuels d'identification des relations implicites. Dans les sections suivantes, nous revenons plus en détail sur ces systèmes en présentant d'abord les difficultés liées à la tâche puis les approches existantes.

3.2.2 Un problème complexe

Nous avons présenté dans le chapitre précédent, notamment en section 2.4.2, les indices autres que les connecteurs discursifs qui avaient pu être mis à jour dans la littérature sur l'analyse discursive. Rappelons que le but d'un système d'identification des relations est de classer des paires de segments textuels, c'est-à-dire de leur attribuer une étiquette parmi un jeu de relations pré-définies correspondant à des liens que l'on a pu regrouper en quatre grandes classes : *Temporal*, *Contingency*, *Comparison* et *Expansion*. Ces classes sont ensuite généralement raffinées en sous-types prenant en compte la direction de la relation (*Result vs Explanation*) et des effets ou contraintes différents (*Contrast vs Concession* ou *veridical vs non veridical*). On suppose ici que l'on est dans le cadre des relations implicites sinon, comme on l'a vu, le connecteur est un indice presque suffisant. On part donc de deux séquences de mots et il faut découvrir le lien qui les unit.

Étant donnée la nature sémantico-pragmatique des liens à inférer, on devrait idéalement disposer de nombreuses ressources et de bases de connaissances complexes. En effet, nous avons vu que ces indices étaient de nature variée. Si les informations lexicales comme les lexicalisations alternatives

10. Nous ne sommes pas sûre du nombre de connecteurs utilisés : le lexique DimLex (STEDE et UMBACH, 1998) contient actuellement 290 formes, mais il est précisé sur le site de téléchargement du lexique (`https://github.com/discourse-lab/dimlex`) qu'il a été enrichi récemment d'une centaine de formes, et il est possible que d'autres ajouts soient intervenus entre l'étude de 2004 et ces additions récentes.

et certains lexèmes, comme « *suppose* » qui peut ancrer un arbre discursif en D-LTAG, peuvent paraître relativement simples à modéliser, même si nous verrons que le problème d'éparpillement des données rend l'estimation des paramètres associés difficile, il n'en est pas de même pour tous les indices. Ainsi, certains requièrent déjà des informations morpho-syntaxiques, comme les marques morphologiques, temps ou mode des verbes, un participe présent pouvant indiquer une relation causale, ou nombre et genre des noms, des oppositions à ce niveau pouvant signaler un contraste, ou la présence de modaux comme *should, can*... en anglais. Il faut donc disposer d'outils de prétraitements et/ou définir des heuristiques capables de fournir ce genre d'informations. De plus, les systèmes automatiques ont montré l'importance de l'analyse syntaxique pour la tâche. Le système de référence le plus performant à l'heure actuelle est fondé sur les règles syntaxiques mises en jeu dans les deux segments. Notons que les études récentes mettent aussi en jeu d'autres éléments discursifs comme la coréférence ou la structuration en topique, informations qui requièrent en général le développement de systèmes dédiés.

Nous avons vu que l'inférence des relations pouvait aussi venir d'indices sémantiques et de connaissances partagées. Ainsi, on a pu mettre en avant des informations de polarité et des relations de hiérarchie entre éventualités. Modéliser ces informations nécessite des ressources encodant ces caractéristiques pour les mots d'une langue. Pour l'anglais, on a pu utiliser des lexiques reflétant la subjectivité des mots et les classes de Levin pour catégoriser les verbes (Pitler et al., 2009), ou des ressources comme Wordnet (Sporleder, 2008 ; Biran et McKeown, 2013).

À notre connaissance, aucune étude n'a comparé les différentes ressources existantes pour les indices comme la polarité, les types de verbe ou les catégories sémantiques et il est probable que des informations supplémentaires puissent venir d'encodages différents de ce type d'informations. De plus, ces dernières années, peu d'études ont introduit de nouveaux traits dans les systèmes, à l'exception de la coréférence. Il reste de nombreuses pistes à explorer à ce niveau, comme l'utilisation de ressources type FrameNet ou des annotations du TimeBank et du PropBank qui recoupent partiellement celles du PDTB. Le réseau Wordnet a été utilisé dans deux études (Sporleder, 2008 ; Biran et McKeown, 2013) cependant sans grand succès mais il est possible qu'une étude plus complète en montre les bénéfices. Bien sûr, ce type de ressources n'est pas disponible pour de nombreuses langues et le mouvement actuel est plutôt vers une diminution du nombre de ressources nécessaires, courant dans lequel s'inscrit aussi notre travail. Cependant il nous semble qu'une large étude empirique reste à faire sur cette tâche afin au moins d'identifier clairement la nature et l'impact des différents types d'indices que l'on veut modéliser et peut-être d'informer les cadres théoriques.

La variété des indices à considérer rend la tâche d'identification des relations implicites complexes. Nous avons décrit les informations à considérer et évoqué la façon dont ces informations avaient pu être prises en compte dans les systèmes automatiques, nous reviendrons plus en détail sur ce point en section 3.3. Cette diversité d'indices participe de la complexité de la tâche parce qu'elle implique de trouver des façons de modéliser finement les données mais aussi parce que, dans le cadre d'un système statistique, elle implique de disposer d'un nombre assez large de données.

3.2.3 Préliminaires en apprentissage statistique

Nous l'avons dit en introduction, un algorithme de classification associe à un objet une classe. Plus formellement, il construit une fonction d'hypothèse associant à chaque objet en entrée du système — une instance, généralement un vecteur —, une étiquette numérique représentant une classe — une valeur discrète parmi un ensemble fini de valeurs représentant des classes d'objets. Cette fonction est utilisée pour faire des prédictions : le but de l'apprentissage est de construire une

fonction dont les prédictions sont les plus proches de la classe correcte. Notons qu'il existe aussi des algorithmes pouvant associer plusieurs classes, ou un ensemble de classes, à une instance, c'est le cadre de la classification *multi-label*. Ce genre d'algorithme est par exemple utilisé en classification de documents où un document peut appartenir à plusieurs classes, par exemple la politique et la finance.

Nous avons également dit que les approches en apprentissage automatique peuvent être groupées en deux grandes familles. L'apprentissage *supervisé* correspond au cas où l'on cherche à inférer une fonction de correspondance entre les instances et les étiquettes, dans notre cas une fonction de classification. Cette fonction est apprise à partir d'instances associées à leur étiquette, les *données d'entraînement*. En apprentissage *non supervisé*, on ne dispose que de données non étiquetées : le but est donc d'identifier des régularités dans ces données à partir desquelles on peut notamment déduire des groupements (ou *clusters*) de données similaires ou proches. La frontière entre ces deux familles est cependant poreuse, on peut faire varier le degré de supervision d'un système. On parle généralement d'apprentissage *semi-supervisé* (ZHU, 2005) lorsque le système se fonde sur des données annotées et non annotées, les secondes étant généralement disponibles en bien plus grande quantité — on peut alors par exemple chercher à propager les étiquettes disponibles à travers l'ensemble des données (*label propagation*) (ZHU et GHAHRAMANI, 2002), on tire en fait profit des données non étiquetées pour densifier l'espace, pour rendre les frontières entre les classes moins denses et donc plus faciles à identifier — ou utiliser un modèle d'étiquetage préalablement construit, éventuellement à partir de règles écrites à la main, comme c'est le cas par exemple des techniques dites d'*auto-apprentissage* (*self learning*) (YAROWSKY, 1995). De manière générale, la tâche de classification des relations discursives s'inscrit dans un cadre supervisé.

3.2.3.1 Classifieur linéaire

La modélisation des données consiste à construire une fonction de représentation qui associe à chaque observation — comme la paire d'arguments d'une relation — une représentation — ici, un vecteur à valeurs réelles. Rappelons que l'on appelle *trait* (ou descripteurs) les coordonnées ou composantes de ce vecteur. Dans cette thèse, une instance est représentée par un vecteur $\mathbf{x} \in \mathbb{R}^d$, d étant le nombre de traits. Parmi les différents types de classifieurs développés pour la classification, les classifieurs *linéaires* construisent une fonction de prédiction qui correspond à une combinaison linéaire sur les composantes du vecteur d'instance considéré, ce qui implique que les interactions entre les traits ne sont pas prises en compte. Cette fonction met en jeu un *vecteur de paramètres* ou *vecteur de poids* $\mathbf{w} \in \mathbb{R}^d$, le poids pour une composante étant calculé indépendamment des valeurs des autres dimensions. Pour un problème de classification binaire (i.e. deux classes), un classifieur linéaire construit un hyperplan (surface de décision, *decision boundary*) séparant l'espace en deux sous-espaces représentant chacune des classes. Dans un problème à deux dimensions (dans $\mathbb{R}^2$), cet hyperplan est une droite de vecteur orthogonal $\mathbf{w}$ comme illustrée dans la figure 3.1. Dans cette illustration, le classifieur a appris une règle de décision qui sépare parfaitement les instances de chacune des classes, représentées par des ronds noirs et blancs. Dans le cas général, on cherche l'hyperplan orthogonal à $\mathbf{w}$ de $d-1$ dimensions défini par l'équation $\mathbf{w} \cdot \mathbf{x} + b = 0$ dans $\mathbb{R}^d$, où b est le biais, donnant la déviation de l'hyperplan par rapport à l'origine. En intégrant le biais b au vecteur $\mathbf{w}$, on ajoute alors une composante aussi à $\mathbf{x}$, l'équation se réécrit $\mathbf{w} \cdot \mathbf{x}$ dans $\mathbb{R}^{d+1}$ et définit un hyperplan à d dimensions. Nous décrivons les façons de modéliser le cas multiclasse dans la section suivante 3.2.4.

Dans le cadre d'un apprentissage supervisé, le vecteur de poids est estimé à partir des données d'entraînement en optimisant un certain critère, par exemple en maximisant la vraisemblance des données ou en minimisant le taux d'erreur. En général, un algorithme met en jeu ce que nous

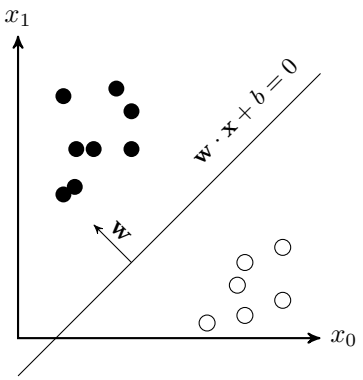


Figure 3.1.: Illustration d'un classifieur linéaire séparant l'espace en deux classes.

appellerons des hyper-paramètres, par exemple un coefficient de régularisation (lissage des valeurs pour les attributs peu fréquents avec Naïf Bayes pour éviter des probabilités nulles, coefficient permettant de forcer des poids petits avec un algorithme par régression logistique pour éviter une trop grande attache aux données) ou un nombre d'itérations (par exemple pour le perceptron) etc. . .. L'optimisation des paramètres du modèle (i.e. le vecteur de poids) et des valeurs des hyper-paramètres se fait souvent à partir d'un ensemble de données dites de *développement* : on choisit le modèle (ensemble des paramètres et des hyper-paramètres) qui obtient les meilleures performances (pour une mesure d'évaluation choisie au préalable) sur ces données différentes de l'ensemble d'évaluation. On utilise finalement le meilleur modèle sur les données d'évaluation afin de donner les performances finales du système.

3.2.3.2 Évaluation

L'évaluation d'un système de classification se fait à partir de données, dites *données d'évaluation*, similaires aux données d'entraînement/développement — de même nature et mettant en jeu les mêmes étiquettes — mais différentes — c'est-à-dire non vues à l'entraînement/développement, ce qui permet de tester la capacité de généralisation du modèle à des données inconnues.

Correct/Prédit	-1	1
-1	TN	FP
1	FN	TP

Table 3.1.: Matrice de confusion en binaire : valeurs utilisées pour définir les mesures d'évaluation.

Nous rappelons ici brièvement la définition des différentes mesures d'évaluation utilisées pour analyser un système de classification. Pour chaque exemple de l'ensemble de données d'évaluation associé à une étiquette correcte, l'algorithme renvoie une étiquette prédite. On peut alors construire une matrice de confusion qui contient le compte des échecs et réussites de l'algorithme. Plus précisément, la matrice de confusion pour un problème binaire contient quatre zones, que l'on peut voir représenter dans le tableau 3.1 :

— Vrai positif (TP) : nombre d'exemples dont l'étiquette correcte est l'étiquette positive et l'étiquette prédite est l'étiquette positive (exemples positifs correctement prédits)

— Faux négatif (FN) : nombre d'exemples dont l'étiquette correcte est l'étiquette positive et l'étiquette prédite est l'étiquette négative (exemples positifs incorrectement prédits)

— Vrai négatif (TN) : nombre d'exemples dont l'étiquette correcte est l'étiquette négative et l'étiquette prédite est l'étiquette négative (exemples négatifs correctement prédits)

— Faux positif (FP) : nombre d'exemples dont l'étiquette correcte est l'étiquette négative et l'étiquette prédite est l'étiquette positive (exemples négatifs incorrectement prédits)

On peut alors définir les mesures de précision (P), de rappel (R) et de f-mesure (F), que l'on calcule pour une classe c. Dans le cas binaire, on donne en général les scores pour la classe (choisie comme) positive (dans notre cas, on pourra construire des classifieurs binaires discriminant une classe, une relation r, par rapport à toutes les autres, le score rapporté correspondra alors à celui de la relation r). Ces mesures sont définies de la façon suivante :

$$P(c) = \frac{TP}{TP + FN} \; ; \; R(c) = \frac{TP}{TP + FP}$$

$$F_\beta(c) = \frac{(1 + \beta^2) \times \text{prec}(c) \times \text{rec}(c)}{\beta^2 \times \text{prec}(c) + \text{rec}(c)} \; ; \; F_1(c) = \frac{2 \times \text{prec}(c) \times \text{rec}(c)}{\text{prec}(c) + \text{rec}(c)}$$

La f-mesure est la moyenne harmonique pondérée selon un coefficient β de la précision et du rappel. On utilise généralement la F_1 ($\beta = 1$) car la précision et le rappel sont alors pondérés de façon égale. Cette mesure permet de rendre compte des performances générales du système. Pour le cas multiclasse, on peut toujours calculer les mesures de précision, rappel et F_1 par classe.

Pour donner des performances globales, on peut également effectuer une moyenne sur les classes pour ces différentes mesures. On distingue micro- et macro-moyenne, on calcule la moyenne pour un score S pour un problème avec k classes par (VAN ASCH, 2013) :

— $S_{\text{micro}} = S(\sum_{c=1}^{k} \text{TP}_c, \sum_{c=1}^{k} \text{FP}_c, \sum_{c=1}^{k} \text{FN}_c, \sum_{c=1}^{k} \text{TN}_c)$

— $S_{\text{macro}} = \frac{1}{k} \sum_{c=1}^{k} S(\text{TP}_c, \text{FP}_c, \text{FN}_c, \text{TN}_c)$

La micro-précision est en fait égale au micro-rappel (et donc à la micro-F_1) [11], on parle généralement d'exactitude micro-moyennée (*micro-averaged accuracy*), ou simplement exactitude, elle correspond au nombre d'exemples bien classés ($TP + TN$) par rapport au nombre total d'exemples ($TP + TN + FP + FN$) :

$$\text{exactitude}_{\text{bin}} = \frac{TP + TN}{TP + TN + FP + FN}$$

Cette mesure est moins souvent utilisée dans le cas binaire où l'on est plus intéressé par les performances du système pour la classe positive, surtout dans le cas de classes déséquilibrées. Elle est par contre très utilisée dans le cas multiclasse où l'on ne distingue pas une classe par rapport aux autres. Dans un problème multiclasse (mc) à k classes, l'exactitude du système évalué sur n exemples est :

$$\text{exactitude}_{\text{mc}} = \frac{\sum_{c=1}^{k} TP(c)}{n}$$

Les macro-moyennes sont également plus souvent utilisées pour des systèmes multiclasses. Elles correspondent à une moyenne pondérée par le nombre de classes, et sont donc plus aptes à rendre compte des performances du système dans le cas de classes déséquilibrées. Les scores macro-moyennés pour un problème à k classes correspondent aux valeurs :

$$\text{macro-prec} = \frac{1}{k} \sum_{c=1}^{k} P(c) \; ; \; \text{macro-rec} = \frac{1}{k} \sum_{c=1}^{k} R(c) \; ; \; \text{macro-F}_1 = \frac{1}{k} \sum_{c=1}^{k} F_1(c)$$

11. En effet, en binaire, pour la classe négative, les vrais positifs correspondent à la valeur TN et les faux positifs à la valeur FN, on a donc $\sum_{c=1}^{k} \text{TP}_c = TP + TN$ et $\sum_{c=1}^{k} \text{FP}_c = FP + FN$. On peut généraliser cette remarque au cas multiclasse. Ce n'est pas le cas si le système n'assigne pas une classe à chaque instance

3.2.4 Difficultés liées à l'apprentissage automatique

3.2.4.1 Classification multiclasse

La tâche d'identification des relations discursives correspond à un problème de classification multiclasse. L'apprentissage multiclasse, où le nombre de concepts cibles $C > 2$, est réputé plus difficile que l'apprentissage binaire, où $C = 2$, puisque l'on doit apprendre plusieurs surfaces de séparation. Il existe différentes possibilités pour résoudre un problème multiclasse qui impliquent en général d'entraîner des classifieurs binaires ou d'optimiser un problème plus complexe. On peut distinguer deux grandes catégories d'algorithmes multiclasse, nous reprenons la terminologie de MOHRI et al. (2012).

Les *algorithmes agrégés* correspondent à la formulation d'un problème multiclasse en problèmes binaires, c'est-à-dire à la combinaison de plusieurs classifieurs binaires pour obtenir un système multiclasse en utilisant n'importe quel algorithme pour construire les classifieurs binaires. Dans ce cadre, on peut adopter trois types de stratégies pour lesquelles on apprend plusieurs classifieurs binaires indépendamment les uns des autres. La première stratégie consiste à apprendre pour chaque classe un classifieur qui la discrimine par rapport à toutes les autres classes, stratégie dite *one-vs-all* ou *one-vs-rest*. Cela nécessite donc la construction de C classifieurs. La prédiction se fait généralement en choisissant la classe ayant obtenu le score le plus haut. Si on note f_i le classifieur pour la classe i, $0 \leq i \leq C$, la fonction de prédiction pour une instance x en entrée du système peut s'écrire : $f(x) = \text{argmax}_i\ f_i(x)$. Pour la seconde méthode, on cherche à discriminer les classes deux à deux, on appelle cette stratégie *one-vs-one* ou *all-vs-all*. On construit alors $C(C-1)$ classifieurs (ou $\frac{C(C-1)}{2}$ si on prend en compte un seul classifieur par paire de classes), donc, généralement, plus de classifieurs que pour la stratégie précédente mais entraînés sur des données moins nombreuses. La prédiction se fait en faisant voter les différents classifieurs. Si on note $f_{i,j}$ le classifieur où i correspond à la classe positive et j la classe négative (avec $f_{j,i} = -f_{i,j}$), la prédiction peut se faire en utilisant la fonction $f(x) = \text{argmax}_i\ (\sum_i\ f_{i,j}(x))$. Une autre solution, appelée *code correcteur d'erreur* (*Error Correcting Code*) consiste à assigner à chaque classe un code sous la forme d'un vecteur, originellement binaire : la tâche est alors de prédire un code et on assigne à l'instance la classe dont le code est le plus proche de celui qui a été prédit (DIETTERICH et BAKIRI, 1995 ; ALLWEIN et al., 2000).

Les *algorithmes non combinés* sont originellement conçus pour les problèmes multiclasse. C'est le cas de l'algorithme naïf bayésien ou des classifieurs par arbre de décision. On peut également proposer une formulation multiclasse d'un algorithme conçu pour le cas binaire. Par exemple, l'algorithme à vecteurs de support (SVM) a été adapté en passant par une modification du vecteur de paramètres qui devient une matrice contenant un vecteur de paramètres par classe, transformation similaire à la configuration *multi-prototype* proposée par CRAMMER et al. (2006) pour l'algorithme Passive Aggressive. Pour ce dernier algorithme, notons cependant que le passage au multiclasse de CRAMMER et al. (2006) passe en fait par une formulation binaire puisque le problème d'optimisation porte uniquement sur la classe prédite et la classe à prédire, laissant inchangés les paramètres pour les autres classes. Une extension exacte au cas multiclasse pour cet algorithme a été proposée par MATSUSHIMA et al. (2010). Pour l'algorithme de régression logistique, on appelle généralement *multinomial* ou *softmax* la version correspondant à une version multiclasse qui est cependant assez rarement utilisée (KRISHNAPURAM et al., 2005).

La supériorité de l'une ou l'autre de ces méthodes n'est pas clairement établie, l'utilisation d'une simple stratégie comme *one-vs-all* pouvant conduire à des performances équivalentes à celles obtenues avec des méthodes plus complexes (RIFKIN et KLAUTAU, 2004), et chacune de ces stratégies pose des problèmes spécifiques (MOHRI et al., 2012). Les études sur l'identification

des relations implicites ont mis en jeu différents algorithmes parmi lesquels l'algorithme naïf bayésien est probablement le plus répandu. Plusieurs auteurs ont effet trouvé que cet algorithme permettait d'obtenir de meilleures performances, du moins pour des classifieurs binaires (PITLER et al., 2009 ; RUTHERFORD et XUE, 2014). Lorsque la tâche est formulée en un problème multiclasse, on a plus rarement une comparaison d'algorithmes et la stratégie multiclasse n'est pas souvent précisée : RUTHERFORD et XUE (2015) et LIN et al. (2009) utilisent un algorithme par régression logistique, WANG et al. (2012) utilisent l'algorithme naïf bayésien et un algorithme par arbre de décision, le second se montrant généralement plus performant. Enfin, notons que LI et NENKOVA (2014a) comparent un algorithme de type SVM et une stratégie de vote de multiple classifieurs binaires mais, comme nous le détaillerons dans la section 3.3.3, la comparaison ne nous semble ni très juste ni très claire.

3.2.4.2 Déséquilibre des classes

Les données implicites sont fortement déséquilibrées en termes de nombre d'exemples par classe, problème connu sous le terme de *déséquilibre des classes*. Dans ce cadre, les algorithmes de classification ont tendance à favoriser la ou les classes majoritaires et on obtient généralement des performances basses pour les classes peu représentées. Dans le cas extrême, le classifieur prédit tous les exemples vers la classe majoritaire ce qui peut correspondre à des performances qui semblent bonnes selon le score considéré.

Il existe différents types de déséquilibre (HE et GARCIA, 2009). Notamment, on peut considérer que l'on a affaire à un véritable déséquilibre entre les classes quand la proportion est de l'ordre de 100 à 10 000 exemples dans la classe majoritaire contre 1 dans la classe minoritaire. Dans notre cas, on a plutôt un déséquilibre dit relatif : les instances des classes minoritaires sont rares relativement à la classe majoritaire. Selon HE et GARCIA (2009), la classe minoritaire peut être relativement bien apprise dans le cas de déséquilibre relatif, ce qui suggère que le degré de déséquilibre n'est pas le seul facteur qui pose problème : la complexité des données est le premier facteur déterminant de la baisse de performance amplifiée par le déséquilibre. De plus, notons que les classes du PDTB correspondent à plusieurs sous-types eux-mêmes déséquilibrés. Cette situation correspond au déséquilibre intra-classe. A l'intérieur des classes, et notamment des classes minoritaires, on peut avoir des sous-concepts sous-représentés et qui seront donc difficiles à modéliser et à différencier d'une forme de bruit au sein du concept principal.

Différentes méthodes ont été proposées pour gérer le problème de déséquilibre des classes. Nous nous limitons ici à la description des stratégies mises en place dans les études sur l'identification des relations implicites, une bonne description des difficultés engendrées par cette configuration et des solutions existantes peut être trouvée dans (HE et GARCIA, 2009 ; HE et MA, 2013 ; WEISS, 2013). En théorie dans ce cas il faudrait pondérer les probabilités en sortie selon la distribution originelle pour reconstruire les probabilités correctes. En pratique, on ignore généralement cette pondération puisque cette modification permet d'améliorer les performances sur les classes rares généralement considérées comme plus importantes (STORKEY, 2009).

La première stratégie, la plus simple, consiste à agir directement au niveau des données. Elle consiste en un rééchantillonnage des exemples permettant de rééquilibrer les données, c'est-à-dire d'obtenir un corpus d'entraînement où toutes les classes sont représentées par le même nombre d'exemples. En général, dans le cas d'un classifieur binaire, la classe majoritaire est la classe négative, il semble rarement être envisagé d'avoir une classe positive majoritaire. Le sur-échantillonnage aléatoire (*random oversampling*) consiste à répliquer certains exemples de la classe minoritaire de manière aléatoire jusqu'à obtenir autant d'exemples que dans la classe majoritaire. Cette méthode mène potentiellement au problème de sur-entraînement car en dupliquant les exemples, on insiste sur

certaines caractéristiques des données auxquelles le classifieur finit par accorder trop d'importance. Le sous-échantillonnage aléatoire (*random undersampling* ou *downsampling*) consiste à supprimer des exemples de la classe majoritaire de manière aléatoire. Ceci peut conduire à rater certaines caractéristiques importantes de la classe majoritaire donc à obtenir une séparation des classes moins pertinente.

La deuxième stratégie envisagée dans la littérature consiste à attribuer un coût différent aux exemples proportionnel au nombre d'exemples dans la classe. Plus précisément, on définit un coût pour la prédiction erronée d'un exemple de la classe majoritaire en tant qu'exemple de la classe minoritaire et un coût pour le cas inverse. Le coût d'une erreur pour un exemple de la classe minoritaire est plus élevé que pour l'erreur inverse. L'objectif de l'apprentissage est alors de minimiser le coût total. On peut intégrer cette fonction de coût de diverses façons. Pour notre tâche, ce coût est intégré à la fonction objective de l'algorithme (Li et Nenkova, 2014a ; Rutherford et Xue, 2014 ; Rutherford et Xue, 2015). Par exemple, Rutherford et Xue (2015) utilisent un algorithme de régression logistique et pondèrent les instances en utilisant le poids suivant pour une instance i de la classe j avec n le nombre total d'instances dans le corpus et $C = \{c_1, \ldots, c_j, \ldots, c_{|C|}\}$ l'ensemble de classes :

$$w_{i,j} = \frac{n}{|C| \times |c_j|}$$

Ainsi, dans un problème où une classe correspond à 10 exemples et la seconde classe à 100 exemples, les exemples de la première recevront un poids de $5,5$ et ceux de la seconde un poids de $0,55$ reflétant le fait qu'une erreur sur la classe minoritaire coûte 10 fois plus cher qu'une erreur sur la classe majoritaire forçant ainsi l'algorithme à porter plus d'attention aux exemples rares. La somme des poids pour toutes les instances d'une classe est la même pour toutes les classes : $\frac{n}{|C|}$. Notons que le processus ressemble au sur-échantillonnage aléatoire, mais sans nécessiter de duplication d'exemples, donc un entraînement plus long, et en permettant une optimisation qui reflète l'importance que l'on accorde à certains exemples.

Dans le cadre de données déséquilibrées, il faut porter une attention toute particulière à la mesure d'évaluation utilisée comme souligné dans (He et Garcia, 2009). L'exactitude micro-moyennée ou plus simplement (micro-)exactitude, en particulier, peut donner une fausse idée des performances du classifieur. Dans le cas où le classifieur prédit tous les exemples comme appartenant à la classe majoritaire, l'exactitude correspond exactement à la proportion d'exemples de la classe majoritaire dans les données, système de référence dit de majorité. Même si cette valeur peut être élevée, un classifieur ayant ce comportement n'est généralement pas souhaitable. Nous verrons dans la section suivante que cette mesure est pourtant souvent rapportée. En général, elle n'est cependant pas la seule mesure donnée. Dans le cas de classifieurs binaires, la F_1, généralement rapportée, est déjà une meilleure métrique puisqu'elle correspond aux performances sur la classe positive, donc minoritaire, et fournit de plus un score global en combinant précision et rappel. Pour le cas multiclasse, les mesures macro-moyennées (macro-F_1 macro-précision et macro-rappel), définies comme la somme des mesures (respectivement F_1, précision et rappel) pour chaque classe pondérée par le nombre de classes, fournissent une alternative convenable à l'exactitude.

3.2.4.3 Éparpillement des données

L'éparpillement des données est un problème connu en TAL qui correspond à deux problèmes liés : on ne dispose que d'un nombre limité de données pour un problème complexe et les données sont représentées dans un espace de large dimensionnalité. La représentation des données dans un espace de grande dimensionnalité pose problème dans le sens où le nombre de données nécessaire

augmente avec le nombre de paramètres, on doit pouvoir observer chaque dimension suffisamment souvent, avoir suffisamment d'exemples avec les différentes combinaisons de dimensions possibles pour construire une bonne estimation des paramètres. Ce problème est connu sous le nom de « malédiction de la dimensionnalité ». Plus le nombre de dimensions est important, plus il y a d'associations possibles à prendre en compte, donc une plus grande combinatoire de variables, et plus il faudra d'exemples pour les rencontrer suffisamment souvent. Elle pose aussi bien sûr problème au niveau computationnel puisque l'entraînement d'un modèle complexe est plus coûteux. Le fait d'avoir des données de grande dimensionnalité, pour un nombre fini d'exemples, rend la représentation éparpillée au sens où certains phénomènes sont rarement observés mais aussi parce que la représentation vectorielle contient plus de valeurs nulles que de valeurs non nulles. La rareté des données entraîne une difficulté de généralisation, parce que le modèle aura du mal à construire une estimation correcte mais aussi parce qu'il ne pourra pas gérer les dimensions qu'il n'aura pas vues à l'entraînement, et peut mener au sur-apprentissage, le modèle apprenant alors des règles trop spécifiques. Le fait d'avoir une représentation vectorielle contenant plus de valeurs nulles que non nulles rend aussi difficile la généralisation car la distance entre les points est très grande, il est donc difficile de construire une surface de séparation, celle-ci passant normalement entre les zones denses de l'espace. Ce problème apparaît notamment avec les traits fondés sur les mots en TAL comme les n-grammes ou, dans notre cas, les paires de mots. Les solutions proposées pour notre tâche consistent à augmenter le volume des données, afin de combattre la rareté, ou à densifier l'espace vectoriel, ce qui a pour effet de combattre à la fois la rareté et le problème dimensionnel puisque cette transformation aboutit généralement à une représentation dans un espace de dimension réduite.

3.3 Études précédentes

Nous avons vu dans les sections précédentes que la tâche d'identification des relations implicites était à la fois cruciale et difficile. La difficulté tient notamment dans la diversité des indices mis en jeu et le phénomène d'éparpillement qu'ils engendrent. Les différentes études sur cette tâche, qui sont sur le PDTB à l'exception des plus anciennes, ont apporté différentes réponses pour gérer ces problèmes. Avant de présenter les études existantes sur cette tâche, nous voulons faire un point en section 3.3.1 sur différents choix faits dans les études dont certains rendent difficiles des comparaisons directes entre elles. Nous listons ensuite en section 3.3.2 les traits les plus couramment utilisés pour notre tâche afin de pouvoir y faire référence par la suite. Les études existantes peuvent être organisées en deux courants. Une partie de ces études, décrites en section 3.3.3, se fonde majoritairement sur des stratégies entièrement supervisées visant à l'identification et à l'extraction des indices pertinents. Ces stratégies présentent certaines limites qui ont conduit au développement de méthodes reposant sur une forme de non-supervision décrites en section 3.3.4. Notre travail s'inscrit également dans ce cadre général qui s'est d'abord instancié à travers l'utilisation de données annotées automatiquement à partir des connecteurs (3.3.4.1), avant de revenir au problème de la représentation des données en proposant de la transformer à partir de ressources acquises automatiquement (3.3.4.2).

3.3.1 Configurations et problèmes de comparaison entre les études

Dans la section précédente, nous avons évoqué les difficultés posées par la modélisation de notre tâche en un problème d'apprentissage statistique, difficultés qui ont pu être prises en compte de manière différente dans les études existantes : stratégie de classification multiclasse, prise en compte du déséquilibre des classes et mesures d'évaluation. La question de l'éparpillement des données, plus générale, sera abordée dans la description des études. De plus, nous avons indiqué

dans le chapitre précédent certaines spécificités du PDTB, comme les différents types de relations ou la possibilité d'annotations multiples. Nous revenons ici sur ces points qui engendrent différents problèmes de comparaison entre les études existantes. Une autre différence, non encore évoquée, vient des données utilisées à l'évaluation. Nous nous focalisons ici sur les études sur le PDTB.

3.3.1.1 Classification multiclasse *vs* binaire

Un système de classification automatique des relations implicites doit identifier un lien parmi un ensemble pré-établi comportant plus de deux relations. On a donc un problème multiclasse. Nous avons dit que c'était une configuration plus difficile que l'apprentissage binaire. Cette difficulté a généralement été contournée dans les études sur le PDTB, et, assez bizarrement, seulement, ou presque, pour celles opérant au premier niveau de la hiérarchie de relation. On a donc ici une différence majeure entre les études au niveau 1 et celles au niveau 2 : pour le premier, on construit généralement un classifieur binaire par relation tandis que pour le second, on construit un classifieur multiclasse.

A notre connaissance, la première étude ayant utilisé les données du PDTB est celle de Blair-Goldensohn et al. (2007). Dans cette étude, trois relations sont considérées dont l'une correspond à l'absence de relation. Les auteurs modélisent le problème à l'aide de trois classifieurs binaires chacun opposant une relation à une autre, les ensembles d'entraînement et d'évaluation sont équilibrés en classe. Il est clair que cette configuration n'est pas très réaliste, on ne peut l'appliquer à un ensemble d'évaluation naturel contenant des exemples de toutes les classes dans leur distribution originelle. Après cette étude, un modèle s'est imposé, celui proposé dans (Pitler et al., 2009). Considérant l'approche précédente, qui est aussi celle de (Marcu et Echihabi, 2002), comme peu réaliste, les auteurs proposent de modéliser le problème à travers plusieurs classifieurs binaires mais avec une stratégie de type *one-vs-rest*. Les ensembles d'entraînement pour chaque classifieur sont constitués en formant la classe positive avec les exemples de la relation considérée et la classe négative avec les exemples des autres relations. Comme nous allons le voir, les ensembles d'entraînement sont équilibrés mais pas l'ensemble d'évaluation ce qui correspond à une configuration plus réaliste. Le fait de conserver des classifieurs binaires n'est pas justifié mais relève probablement du problème évoqué précédemment concernant la difficulté de la classification multiclasse.

Toutes les études suivantes au niveau 1 de sens ont suivi cette configuration jusqu'à récemment (Li et Nenkova, 2014a ; Rutherford et Xue, 2015). Au contraire, la première étude au niveau 2 a imposé une modélisation multiclasse (Lin et al., 2009). Notons cependant qu'une étude à ce niveau de sens donne des résultats en binaire uniquement (Lan et al., 2013). Ce point ne pose que peu de problèmes de comparaison entre les études, puisque la plupart font comme les précédentes, mais est assez révélateur d'un manque de consistance général pour cette tâche qui commence cependant à s'inscrire dans un cadre plus rigoureux.

3.3.1.2 Rééchantillonnage *vs* pondération des instances

Le fait d'utiliser des classifieurs binaires ou multiclasses est aussi lié à deux autres problèmes : la gestion du déséquilibre des classes et l'évaluation. Concernant le premier point, Pitler et al. (2009) ont également proposé une stratégie pour le niveau 1 qui a été suivie dans toutes les études jusqu'à très récemment. Nous avons évoqué les différentes façons de gérer ce problème. Pitler et al. (2009) proposent celle correspondant à un sous-échantillonnage aléatoire des exemples majoritaires. C'est une stratégie simple mais ce n'est pas la meilleure stratégie proposée dans la littérature (He et Garcia, 2009). L'un de ses inconvénients est le caractère aléatoire des suppressions ce qui peut engendrer la suppression d'exemples pertinents. En général, deux systèmes constitués à partir

de sous-échantillonnages différents ne conduiront pas aux mêmes résultats, ce qui bien sûr pose problème quand il s'agit de les comparer. Ainsi, PARK et CARDIE (2012) testent une optimisation de l'ensemble d'entraînement constitué à partir de multiples sous-échantillonnages et trouvent d'importantes différences au niveau des performances. Optimiser l'ensemble d'entraînement est cependant un pré-traitement assez lourd. Depuis, LI et NENKOVA (2014a) ont mené une étude sur différentes stratégies pour gérer le déséquilibre des classes en comparant uniquement des systèmes de référence au niveau de la modélisation des données. Ils montrent notamment qu'une méthode fondée sur la pondération des instances directement au niveau de l'algorithme est plus efficace. Pour l'instant, seuls (RUTHERFORD et XUE, 2014 ; RUTHERFORD et XUE, 2015) l'ont mise en œuvre dans un système plus raffiné. Notons qu'au niveau 2, aucune étude ne met en œuvre de stratégie gérant le déséquilibre des classes ce qui permet à l'algorithme d'apprendre la distribution en classes des données mais on observe en général des scores bas pour les classes sous-représentées.

3.3.1.3 Micro moyenne *vs* macro moyenne

Concernant l'évaluation, toutes les études au niveau 1 donnent des scores de F_1 pour les classifieurs binaires. Ensuite les études donnent éventuellement des scores de précision et de rappel, et parfois un score d'exactitude. Selon les études, nous ne sommes pas certaine que ce score soit une vraie exactitude micro-moyennée et pas une précision. La tradition pour les systèmes binaires est en effet plutôt de donner les scores pour la classe positive, donc la précision, plutôt qu'un score global. Clairement, HONG et al. (2012) donnent une mesure d'exactitude micro-moyennée alors que WANG et al. (2012) parlent d'exactitude en donnant la formule de la précision et en se comparant aux scores d'exactitude de (PITLER et al., 2009). Si cette dernière étude utilise bien l'exactitude et non la précision, cela pourrait d'ailleurs expliquer les différences rapportées dans (WANG et al., 2012) que ces auteurs expliquent par une différence en terme de pré-traitements. Cela ne pose pas de vrai problème tant qu'un score de F_1 est également donné ce qui n'est pas le cas par exemple dans (HONG et al., 2012) : notons que les auteurs rapportent une exactitude d'environ 96% pour *Temporal* ce qui correspond à peu près à la proportion d'exemples des autres relations dans l'ensemble de test, donc à la performance d'un classifieur par majorité.

PITLER et al. (2009) comme WANG et al. (2012) ne donnent que le score de micro-exactitude pour des expériences multiclasses. Or, ce n'est pas le score le plus pertinent quand les classes ne sont pas équilibrées puisqu'il a tendance à favoriser les classes sur-représentées. Ainsi, pour le niveau 1, dans la configuration de PITLER et al. (2009), un système par majorité correspond à une micro-exactitude de $65,30\%$ donc plutôt similaire au $65,40\%$ rapporté par ces auteurs. Ce score ne reflète pas les performances du système pour les classes peu représentées. Ne donner que ce score ne permet donc pas d'estimer si un système est meilleur pour certaines classes ou s'il est bon en général. C'est pourtant une tendance assez répandue, également au niveau 2 pour lequel LIN et al. (2009) donnent bien des scores par classe ce qui n'est pas le cas dans une étude plus récente (JI et EISENSTEIN, 2014a). La tendance pourrait cependant s'inverser puisque la dernière étude en date pour le niveau 1 fournit des résultats en multiclasse en présentant à la fois les scores par classe et la micro-exactitude mais également une mesure de F_1 macro-moyennée plus adaptée au cas de déséquilibre des classes (RUTHERFORD et XUE, 2015). On note aussi que LI et NENKOVA (2014a) donnent un score de précision macro-moyennée pour le niveau 1. Bien sûr, la multiplicité des mesures peut poser problème pour effectuer des comparaisons, mais le point le plus important est de fournir des résultats permettant d'évaluer l'apport général du système dans le cas d'un problème multiclasse avec déséquilibre des classes (donc d'utiliser plutôt une macro-moyenne) et l'apport spécifique pour certaines relations (donc des scores par classes).

3.3.1.4 Sections d'évaluation

Avant d'aborder les spécificités du PDTB qui ont pu mener à des différences entre les études, notons qu'un autre problème vient du fait que toutes les études n'utilisent pas le même ensemble de test. Le corpus du PDTB comporte 25 sections notées de 00 à 24. Le manuel du PDTB (PRASAD et al., 2007) recommande l'utilisation des sections 2 à 21 pour l'entraînement, de la section 23 pour l'évaluation et de la section 22 pour le développement, les sections restantes pouvant être utilisées pour élargir l'ensemble de développement. Ces recommandations se fondent sur la tradition établie en analyse syntaxique automatique à partir du PTB. Elles n'ont été suivies qu'au niveau 2 par LIN et al. (2009) et probablement par JI et EISENSTEIN (2014a) qui ne précisent cependant pas cette information. Une section représente assez peu d'exemples pour constituer un ensemble d'évaluation (769 pour la section 23). C'est peut-être la raison pour laquelle PITLER et al. (2009) ont proposé une autre configuration, avec les sections 2 à 20 pour l'entraînement et les sections 21 et 22 pour l'évaluation (1 046 exemples). Cette configuration a été suivie dans presque toutes les études au niveau 1 à l'exception de WANG et al. (2010) et HONG et al. (2012) (entraînement sur 2-22 et test sur 23-24) et de LI et NENKOVA (2014a) et LI et NENKOVA (2014b) (entraînement sur 2-19 et test sur 20-24). Ces dernières proposent cette modification afin de pouvoir profiter d'un ensemble d'évaluation plus large permettant une meilleure estimation de la significativité des résultats. Si l'argument est pertinent, il n'en reste pas moins que cette multiplicité des divisions entraînement-test rend les comparaisons difficiles. Pour le niveau 2, LAN et al. (2013) utilisent également les sections 21-22 pour l'évaluation, ce qui semble confirmer un choix définitif pour cette subdivision quel que soit le niveau.

3.3.1.5 Annotations multiples

Nous l'avons dit, le PDTB présente la particularité de permettre aux annotateurs d'annoter plusieurs relations pour un exemple, jusqu'à deux pour les explicites et les AltLex et jusqu'à quatre pour les implicites. Si ce principe est du point de vue théorique valable (il est reconnu également dans le cadre de la SDRT) il n'en reste pas moins que c'est une situation particulière dans un cadre d'apprentissage statistique. Normalement, une telle configuration fait appel à l'apprentissage dit multi-label et à des algorithmes spécifiques permettant de gérer le fait qu'une même instance peut correspondre à plusieurs classes. Normalement toujours, à l'évaluation, un tel algorithme cherche à vérifier que toutes les classes annotées sont reconnues avec éventuellement des scores de reconnaissance partielle. Cependant, si certaines études sur l'identification des relations ont pris en compte ces annotations multiples, ce n'est jamais dans le cadre d'un tel système. Par ailleurs, ce n'est en fait pas vraiment possible puisque moins de 2% des exemples implicites seulement sont annotés avec plusieurs relations.

La question de la prise en compte ou non des annotations multiples n'est pas toujours claire dans les études. Parfois, nous pouvons nous fonder sur le nombre d'exemples par relation indiqué pour l'inférer. Quand ces informations ne sont pas fournies, le doute subsiste. LIN et al. (2009) et JI et EISENSTEIN (2014a) indiquent clairement que ces annotations sont prises en compte mais ne précisent cependant pas qu'ils en conservent trois (rappelons qu'un seul exemple est annoté avec quatre relations). La stratégie consiste alors à dupliquer les exemples autant de fois qu'ils ont d'étiquettes de relation à l'entraînement. Durant la phase d'évaluation, si l'une des étiquettes est correctement prédite, la prédiction est comptée comme correcte. Le fait de dupliquer des exemples à l'entraînement, donc de présenter à l'algorithme plusieurs fois une même instance mais avec des étiquettes différentes, est plutôt une mauvaise chose d'un point de vue apprentissage puisque l'algorithme est fait pour construire une séparation entre les classes, pour les discriminer. Quant à la technique d'évaluation, elle paraît simplifier un peu le problème. Étant donné le peu d'exemples

concernés, l'impact ne doit cependant pas être très important. Il serait cependant intéressant d'étudier ces cas d'annotation multiple afin de voir s'ils présentent une certaine consistance et si l'une des relations paraît plus intéressante à conserver, par exemple une relation causale plutôt qu'une relation temporelle. Une solution plus brutale, plus simple mais qui nous semble aussi justifiée étant donné le peu d'exemples concernés est proposée dans (WANG et al., 2012) : seule la première annotation est conservée. Comme le phénomène est assez rare, nous pensons qu'il a un impact limité sur les résultats finaux et donc n'empêche pas de comparer les systèmes.

3.3.1.6 Cas des AltLex, NoRel et EntRel

Finalement, la dernière grande source de différences entre les études concerne la prise en compte ou non des différents types de relations non explicites.

De nombreuses études ont inclus les AltLex et les EntRel à l'ensemble des relations implicites, ces dernières recevant l'étiquette *Expansion* (seules les études au niveau 1 ont effectué cette fusion). Pour les AltLex, l'argument est que ces exemples ne contiennent pas de connecteur donc ressemblent à des implicites. Selon nous, le fait que les annotateurs n'aient pas pu insérer un connecteur à cause de la présence d'une certaine expression rend au contraire ces exemples très proches des explicites. Quant aux EntRel, les auteurs proposent de les voir comme des exemples relevant d'un certain type d'*Expansion*. Souvent cependant le seul argument est de pouvoir se comparer aux études existantes. Notons qu'encore une fois, ce point n'est pas toujours clair dans les descriptions des configurations d'expérience. Ces deux types de relations sont au moins inclus dans (PITLER et al., 2009 ; PARK et CARDIE, 2012 ; JI et EISENSTEIN, 2014a). ZHOU et al. (2010) et WANG et al. (2010) incluent les AltLex mais pas les EntRel, ce qui est probablement aussi le cas dans (BIRAN et MCKEOWN, 2013). Notons que WANG et al. (2012) et LIN et al. (2009) précisent clairement qu'ils ne prennent en compte que les « vrais » implicites et que RUTHERFORD et XUE (2014) présentent des expériences incluant ou non EntRel dans la classe *Expansion* ce qui ouvre la voie à une comparaison avec de futures études ne prenant plus ce type de relation en compte. Pour nous, ces exemples constituent une classe à part reposant sur des indices différents ce dont rend compte le fait que le schéma d'annotation en ait fait un type spécifique de relation.

Concernant NoRel, la situation est légèrement différente, les exemples portant cette étiquette ne sont jamais inclus dans le groupe d'exemples non explicites. Notons cependant que PITLER et al. (2009) les incluent dans les exemples négatifs des classifieurs binaires de niveau 1. Plus fréquemment, ces exemples peuvent former une classe supplémentaire parfois élargie aux exemples EntRel ce qui prouve la confusion concernant ce type. PITLER et al. (2009) proposent ainsi un système multiclasse avec six classes, les relations de niveau 1, les EntRel et les NoRel. WANG et al. (2010) et LI et NENKOVA (2014a) font un groupe à part regroupant EntRel et NoRel. Nous pensons que cette classe bien particulière doit être traitée différemment des autres. L'absence de relation correspond normalement à la tâche d'attachement, donc à l'identification de la présence ou non d'un lien entre chaque paire de segments discursifs d'un texte. On pourrait donc considérer éventuellement d'en faire une classe à part dans un système multiclasse, même s'il nous semble plus pertinent de séparer les tâches et de gérer l'attachement à l'aide d'un modèle binaire. Cependant, cette tâche d'attachement ne correspondrait pas aux seuls exemples de NoRel dans le PDTB. En effet, l'étiquette NoRel a été ajoutée pour rendre compte de l'absence de relation au niveau local, entre deux phrases adjacentes non liées explicitement. La tâche complète d'attachement devrait au contraire être globale. Tant que tous les exemples n'auront pas été annotés, il semble difficile de mener la tâche d'attachement sur ce corpus.

Notons pour terminer sur ce point des relations prises en compte que LI et NENKOVA (2014b) apportent une certaine originalité en proposant un système multiclasse au niveau 1 dans lequel, en

plus de regrouper EntRel et Norel dans une classe particulière, la relation *Expansion* est divisée en trois de ses sous-types, ce qui permet d'avoir un déséquilibre moins important entre les classes mais qui rend impossible toute comparaison avec les quelques systèmes multiclasses existants pour ce niveau.

3.3.1.7 Résumé

Nous avons vu que de nombreux points rendaient les études difficilement comparables, différents choix sont possibles quant à la configuration d'un système :

- Classification : binaire ou multiclasse au niveau 1 comme au niveau 2,
- Mesure d'évaluation : scores par classe présents ou non, utilisation d'une mesure micro-moyennée ou macro-moyennée pour le multiclasse,
- Sections d'évaluation : sections 21-22, 23-23 ou 20-24 pour le niveau 1, sections 23 ou 21-22 pour le niveau 2,
- Annotations multiples : duplication des exemples pour les trois premières annotations ou utilisation uniquement de la première annotation,
- AltLex : inclus ou non dans les exemples implicites,
- EntRel : inclus dans les exemples implicites avec l'étiquette *Expansion* (uniquement au niveau 1) ou inclus dans une classe supplémentaire avec NoRel ou formant une classe supplémentaire ou non pris en compte,
- NoRel : correspond à une classe supplémentaire ou inclus dans une classe supplémentaire avec EntRel ou inclus dans les exemples négatifs des classifieurs binaires ou non pris en compte.

Très peu d'études mettent en œuvre dans leur comparaison une reproduction des études précédentes et souvent ne précisent pas les potentielles différences à ces différents niveaux. Tout ceci participe d'un certain flou concernant l'évaluation de la tâche. Les dernières études sont cependant plus rigoureuses au niveau de la description de leur configuration et des comparaisons. Nous présentons dans les sections suivantes les études sur l'identification des relations implicites en commençant par celles fondées sur une stratégie entièrement supervisée reposant sur l'utilisation de nombreuses ressources. Nous décrivons ensuite les études mettant en œuvre une forme de non supervision cherchant, dans un certain sens, à limiter l'utilisation de ces ressources.

3.3.2 Motifs de traits

Nous listons dans cette section les motifs des traits utilisés les plus fréquemment pour notre tâche sous la forme des catégories proposées dans la littérature afin de pouvoir nous y référer ultérieurement. Ces traits proviennent des études de MARCU et ECHIHABI (2002), PITLER et al. (2009) et LIN et al. (2009). Bien sûr, d'autres informations ont été utilisées, nous les présenterons dans la suite de ce chapitre.

Paires de mots Ces traits correspondent aux paires de mots définies dans le produit cartésien sur les arguments, un trait correspond à un mot du premier argument et un mot du second. La valeur du trait correspond à une valeur binaire indiquant la présence de la paire de mots pour l'exemple ou la fréquence de cette paire. Ces traits sont introduits dans (MARCU et ECHIHABI, 2002).

Premier, dernier, trois premiers mots Ces traits encodent les premiers, derniers et trois premiers mots de chaque argument. Ici, l'intuition est qu'ils peuvent correspondre à des lexicalisations

alternatives des relations, ils sont donc spécialement conçus pour une configuration incluant les AltLex. Ils peuvent cependant également contenir des mots comme des dates, des jours ou des mois importants pour les relations temporelles. Ces informations sont binarisées : les traits représentent la présence d'un mot en première ou dernière position ou d'une séquence de trois mots au début de l'un des deux arguments. Ils présentent donc le problème d'éparpillement. Ces traits sont introduits dans (PITLER et al., 2009).

Nombre, pourcentage, dollar Ces traits correspondent au nombre de fois où l'on rencontre une expression dénotant un nombre, un montant en dollar ou un pourcentage dans chaque argument et en croisant les arguments, ce qui peut être un indicateur d'une comparaison. Ces traits sont introduits dans (PITLER et al., 2009).

Règles de production Ces traits correspondent aux règles de production, générées à partie de l'analyse syntaxique en conservant les feuilles. Ces traits sont donc du type « NP $\rightarrow$ DT N » et « DT $\rightarrow$ the ». Plus précisément, trois types de traits binaires sont définis indiquant la présence d'une règle dans chaque argument et dans les deux arguments. Les tags fonctionnels (par exemple « SBJ ») sont supprimés. Ces traits se sont montrés particulièrement pertinents pour la tâche et constituent aujourd'hui, à côté des paires de mots, un jeu de traits de référence. Ces traits sont introduits dans (LIN et al., 2009).

Analyse en dépendance Ces règles correspondent aux règles provenant de l'analyse en dépendance, en conservant également les lexèmes. Ces traits sont donc du type « had $\leftarrow$ nsubj dobj » et « problems $\leftarrow$ det nn advmod ». Les motifs de traits sont les mêmes que pour les règles de production. Ces traits sont introduits dans (LIN et al., 2009).

Verbes Cette catégorie contient différents types d'information sur les verbes. Les verbes sont associés à leur classe de Levin (LEVIN, 1993), ce qui doit permettre d'identifier des verbes similaires. Plus précisément, le jeu de traits inclut le nombre de paires de verbes, chacun provenant d'un des deux arguments, qui appartiennent à la même classe de Levin en ne considérant que la plus haute catégorie, ceci permettant également d'effectuer une généralisation sur les mots. Sont aussi représentées la longueur moyenne des groupes verbaux, indices de la complexité syntaxique, et la catégorie morpho-syntaxique du verbe principal de chaque argument, ce qui inclut, dans l'annotation du PTB, une information temporelle. Ces traits sont introduits dans (PITLER et al., 2009).

Catégories sémantiques Le lexique *General Inquirer* [12] (STONE et KIRSH, 1966) associe à chaque mot un ensemble de catégories sémantiques dont des informations de polarité mais aussi des catégories plus fines comme « *vertue* » et « *vice* » ou « *fall* » et « *rise* ». Il contient aussi une catégorie « Comp » qui correspond à des mots tendant à établir une comparaison (« *optimal* », « *other* », « *supreme* »...). Ces catégories sémantiques sont vues comme un moyen de remédier à l'éparpillement des données. Ainsi, le système n'a plus besoin de voir des paires de mots comme « *pousser,tomber* » mais des catégories plus générales. Les traits correspondent aux catégories sémantiques des verbes et à leur produit cartésien. Ces traits sont introduits dans (PITLER et al., 2009).

Polarité Des informations de polarité sont récupérées à partir du lexique issu d'un corpus d'analyse d'opinion, le *Multi-perspective Question Answering Opinion Corpus* (WILSON et al., 2005). Chaque mot présent dans le lexique correspond à une catégorie entre négatif, positif et neutre. Les

12. `http://www.wjh.harvard.edu/~inquirer/`

traits correspondent au nombre de mots négatifs, positifs, négatifs niés, positifs niés et neutres, ainsi que le produit cartésien de ces polarités. La prise en compte de la négation permet de différencier « *nice* » et « *not nice* ». Elle s'effectue en cherchant si un mot précédent (la fenêtre n'est pas précisée) porte la catégorie *Negate* selon le lexique *General Inquirer*. Les indices de polarité sont censés aider à l'identification des exemples de type contrastif donc mettant en jeu des oppositions. Ces traits sont introduits dans (PITLER et al., 2009).

Modalité Afin de représenter des informations de modalité, des traits binaires encodent la présence ou l'absence de modaux comme « *can* », « *should* » ... dans chaque argument, ils sont probablement identifiés en se fondant sur la catégorie morpho-syntaxique (catégorie « MD » dans le PTB, en liste fermée). Les auteurs indiquent qu'ils ajoutent des traits pour des modaux spécifiques sans préciser lesquels. Ils construisent aussi des paires de ces traits sur les deux arguments. Ces informations doivent permettre notamment d'indiquer des déclarations conditionnelles. Ces traits sont introduits dans (PITLER et al., 2009).

Contexte-Pitler Les informations contextuelles correspondent à la présence d'une relation explicite immédiatement avant ou après les segments considérés et, éventuellement, le connecteur et son sens. Ces traits doivent permettre d'identifier des interactions entre relations, une étude sur les explicites (PITLER et al., 2008) avait ainsi montré que les enchaînements entre relations n'étaient pas aléatoires. Il nous semble cependant que ces traits sont peu réalistes puisqu'ils devraient normalement être construits à partir de connecteurs prédits et dont le sens est également prédit. Ces traits sont introduits dans (PITLER et al., 2009).

Contexte-Lin Les informations contextuelles indiquent si l'exemple précédent ou suivant est enchâssé dans l'argument 1 ou 2 de l'exemple courant et inversement, si des arguments sont partagés entre des exemples consécutifs et, si l'exemple précédent ou suivant est explicite, son connecteur. Ces traits reposent sur une identification manuelle des arguments des connecteurs. Ces traits sont introduits dans LIN et al. (2009).

3.3.3 Stratégies entièrement supervisées

Nous l'avons dit, l'une des difficultés de la tâche d'identification des relations implicites est d'identifier et de modéliser la multitude d'indices possibles. Une première lignée d'études s'est donc consacrée essentiellement à cette question : de quel type d'information a-t-on besoin et comment les modéliser dans le cadre d'un système d'apprentissage automatique ? Comme nous l'avons dit, les études sur les implicites ont généralement proposé des classifieurs binaires pour le niveau 1 et multiclasses pour le niveau 2. Différents algorithmes de classification ont été testés, avec la conclusion, du moins dans les premières études, que le classifieur Naïf Bayes conduisait généralement aux meilleures performances. Notons cependant que les études indiquent rarement avoir optimisé les hyper-paramètres des algorithmes testés, ce qui fausse leur conclusion. Nous présentons dans cette section les études fondée sur des stratégies entièrement supervisées dont la visée principale était de définir une représentation des données pertinente.

L'étude de PITLER et al. (2009) a, nous l'avons vu, défini des configurations qui ont été assez généralement suivies par la suite, elle a aussi proposé une modélisation riche souvent reprise. En plus de développer le premier système d'identification des relations implicites sur le PDTB, cette étude avait deux objectifs : tester la pertinence des traits constitués à partir des paires de mots et, en particulier, quand ces paires proviennent des exemples explicites (MARCU et ECHIHABI, 2002),

et évaluer de nouveaux traits, notamment des traits linguistiquement plus motivés. Concernant le premier objectif, les auteurs proposent d'évaluer quatre configurations :

— Le modèle est construit à partir des paires de mots des exemples à classer, donc les relations implicites annotées dans le PDTB.

— Le modèle est construit à partir des paires de mots du corpus artificiel constitué par MARCU et ECHIHABI (2002), donc des relations explicites annotées automatiquement. Notons que ce corpus a été annoté avec quatre relations du cadre RST, PITLER et al. (2009) en utilisent deux : *Contrast*, dont les exemples sont étiquetés avec *Comparison*, et *Explanation-Evidence*, correspondant ici à *Contingency*. Ce jeu de traits n'est donc testé que pour ces deux relations.

— Le modèle est construit à partir des paires de mots correspondant aux relations explicites du PDTB, une variante non bruitée mais entraînée sur moins de données que le précédent (et, incidemment, disponible pour les quatre relations de niveau 1 du PDTB).

— Le modèle est construit à partir des paires de mots correspondant aux relations implicites annotées dans le PDTB, comme pour la première version, mais les paires sont filtrées : une paire est conservée si elle apparaît dans le corpus artificiel de MARCU et ECHIHABI (2002). Cette variante n'est disponible que pour *Comparison* et *Contingency*.

Concernant le second objectif, PITLER et al. (2009) partent du principe que les indices modélisés en se fondant uniquement sur les mots des arguments dans (MARCU et ECHIHABI, 2002) peuvent être modélisés en utilisant des ressources dédiées. Ces indices correspondent à ceux que nous avons pu décrire en section 2.4.2. L'utilisation de ces ressources permet d'apporter une réponse au problème d'éparpillement des données en permettant en théorie une meilleure généralisation de l'information lexicale. PITLER et al. (2009) utilisent donc des ressources qui associent aux mots des étiquettes sémantiques (**Catégories sémantiques**[13] à partir du *General Inquirer*) ou des informations de polarité (**Polarité** à partir du *MPQA*), introduisent des traits représentant des informations de modalité (**Modalité**) et généralisent l'information verbale grâce aux classes de Levin (**Verbes**). PITLER et al. (2009) ont également proposé de prendre en compte des informations contextuelles (**Context-Pitler**) correspondant aux relations explicites autour de l'exemple considéré.

L'évaluation de ces traits se fait avec quatre classifieurs binaires, correspondant aux quatre classes de niveau 1, et en utilisant la stratégie de sous-échantillonnage à l'entraînement, l'ensemble d'évaluation suit la distribution naturelle des données. Ils évaluent des classifieurs de type Naïf Bayes, régression logistique et AdaBoost (FREUND et SCHAPIRE, 1997), mais ne donnent des résultats que pour le premier, les autres donnant, selon eux, des résultats légèrement inférieurs en termes d'exactitude. Ils ne précisent pas s'ils optimisent les paramètres des classifieurs. Les scores de F_1 et d'exactitude qu'ils donnent pour chaque classifieur binaire et chaque ensemble de traits montrent que ce sont des traits surfaciques qui permettent généralement d'obtenir les meilleures performances, notamment les mots aux frontières (**Premier, dernier, trois premiers mots**) qui peuvent correspondre à des lexicalisations alternatives[14], à des expressions temporelles récurrentes comme les jours de la semaine, les mois etc... De même, les paires de mots permettent, à elles seules, d'obtenir des performances assez hautes, en général proches de celles du meilleur système sauf pour *Expansion*. Concernant les paires de mots provenant de données explicites ou artificielles, il est clair que les performances sont inférieures à celles des systèmes utilisant les paires extraites des données implicites (pour les deux relations considérées), ce qui est attendu et correspond aux différences distributionnelles entre ces deux types de données. Les meilleurs systèmes sont généralement obtenus en combinant des traits de type paire de mots et d'autres types de traits (incluant voire limités aux mots aux frontières), montrant l'importance de l'information lexicale. Ils obtiennent au mieux : $16,76\%$ de F_1 pour *Temporal*, $21,88\%$ pour *Comparison*, $47,13\%$ pour

13. Nous avons présenté plus précisément ces traits, associés aux noms de catégorie repris ici, dans la section 3.3.2.
14. Rappelons que les AltLex, ainsi que les EntRel, sont inclus dans les données pour cette étude.

Contingency et $76,42\%$ pour *Expansion*. Ces scores laissent donc une marge assez large pour des améliorations futures, et sont largement inférieurs aux meilleurs scores obtenus pour les explicites (voir section 3.1.2.3).

PARK et CARDIE (2012) ont ensuite exploré de manière systématique les combinaisons des traits introduits par PITLER et al. (2009) : ils concluent sur l'inutilité des traits de type paires de mots car des performances similaires voire supérieures peuvent être obtenues en utilisant des traits linguistiquement motivés, dont l'extraction repose cependant sur l'existence de nombreuses ressources construites à la main. De plus, ils proposent d'effectuer différentes formes de pré-traitements afin de tirer un meilleur profit de ces ressources, comme la stemmatisation ou racinisation (*stemming*) des mots avant l'utilisation d'une ressource comme le lexique *General Inquirer*. Ces auteurs reviennent également sur la stratégie de sous-échantillonnage. Comme nous l'avons dit, le fait de supprimer des exemples d'entraînement, de manière aléatoire, conduit à une difficulté importante : deux ensembles d'entraînement ne conduisent pas forcément aux mêmes performances, ce qui rend la reproduction de résultats difficile. Ces auteurs proposent d'optimiser l'ensemble d'entraînement en effectuant de multiples sous-échantillonnages aléatoires et en conservant l'ensemble donnant les meilleures performances sur les données de développement. Il s'agit cependant d'une forme de pré-traitement assez coûteuse. De plus, comme la combinatoire possible ne peut être entièrement explorée, il n'est pas certain que cela règle le problème de reproductibilité. Les résultats de niveau 1 obtenus par ces auteurs, toujours avec des classifieurs binaires, ont longtemps constitué l'état de l'art pour la tâche avec des scores de F_1 de : $26,57\%$ pour *Temporal*, $49,82\%$ pour *Contingency*, $31,32\%$ pour *Comparison* et $79,22\%$ pour *Expansion*. Ils ont depuis été dépassés dans des études utilisant une forme de non supervision (RUTHERFORD et XUE, 2014 ; RUTHERFORD et XUE, 2015 ; JI et EISENSTEIN, 2014a).

Après cette étude, ont notamment été introduits par LIN et al. (2009) des traits représentant des informations syntaxiques issues d'une analyse en constituants (**Règles de production**) et en dépendances (**Analyse en dépendances**). Dans les expériences menées par ces auteurs, en multiclasse au niveau 2 de relation (classifieur par régression logistique), les paires de mots prises seules conduisent à des résultats ($30,3\%$ d'exactitude) inférieurs aux règles de production ($36,7\%$) mais meilleurs que les traits de dépendance ($26,0\%$) ou contextuels (**Contexte-Lin**, $28,5\%$). En utilisant tous ces traits, LIN et al. (2009) rapportent $40,2\%$ d'exactitude, score également dépassé dans une étude reposant sur une forme de non supervision (JI et EISENSTEIN, 2014a). Les règles de production conduisent aux meilleures performances lorsqu'un seul groupe de traits est considéré. Ils ont été repris dans pratiquement toutes les études suivantes et continuent de correspondre à un système de référence avec des scores hauts, ce qui est assez étonnant dans le sens où il n'est pas vraiment clair en quoi ces informations aident une tâche pour laquelle les distinctions sont plutôt d'ordre sémantique. Par ailleurs, une meilleure prise en compte de la structure syntaxique à l'aide de noyaux d'arbres ne semble pas apporter de réelles améliorations (WANG et al., 2010) [15]. Rappelons que ces traits incluent une information lexicale, puisque les règles liant les catégories morpho-syntaxiques aux tokens sont incluses, et il a été montré (LI et NENKOVA, 2014b) que supprimer ces règles lexicales, qui sont les principales sources d'éparpillement, avait un impact négatif fort sur les scores. Ceci démontre, encore une fois, l'importance de l'information lexicale. Par ailleurs, LI et NENKOVA (2014b) proposent une formulation des règles de production donnant lieu à moins d'éparpillement : les règles sont cassées c'est-à-dire que l'on remplace « S $\rightarrow$ NP VP » par « S $\rightarrow$ NP » et « S $\rightarrow$ VP ». Cette modification n'entraîne cependant que de légères améliorations. Une façon plus brutale, mais néanmoins plus efficace en termes de performance, de gérer l'éparpillement est d'utiliser un filtre sur les traits. LIN et al. (2009) utilisent un filtre en fréquence de 5 sur les traits les plus épars, comme les paires de mots ou les règles de production, et un filtre fondé sur

15. La configuration très particulière des expériences menées dans cette étude, mêlant tâche d'attachement et d'identification de la relation, explicite et implicite, ne permet pas d'effectuer une comparaison complète.

la valeur d'information mutuelle ce qui permet d'améliorer les performances générales ($35,0\%$ d'exactitude sans sélection contre $40,2\%$ avec sélection). Dans les dernières études (RUTHERFORD et XUE, 2014 ; RUTHERFORD et XUE, 2015), le filtre en fréquence est optimisé sur les données de développement ce qui apporte d'importantes améliorations. Filtrer les traits permet de réduire la complexité d'un modèle et éventuellement d'en améliorer la généralisation en réduisant les risques de sur-entraînement.

Les principales informations ajoutées par la suite concernent la coréférence. LOUIS et al. (2010b) utilisent un sous-ensemble du PDTB annoté également en coréférence dans le corpus Ontonotes (590 documents). Dans le cadre d'un système d'identification des relations entre phrases adjacentes (explicite ou non) modélisé en cinq classifieurs (les quatre classes de niveau 1 et les Entrel), l'utilisation d'informations sur les entités et les formes pronominales n'apportent cependant pas d'amélioration. RUTHERFORD et XUE (2014) incluent également dans leur système des informations sur les entités et les liens de coréférence, en se fondant par contre sur une identification automatique de ces caractéristiques, mais l'étude qu'ils mènent en supprimant les ensembles de traits les plus informatifs incrémentalement démontre que ces traits ont peu d'impact, du moins au niveau 1 de relation. Enfin, JI et EISENSTEIN (2014a) proposent d'inclure ces informations en construisant une représentation par composition vectorielle dans l'arbre syntaxique, le vecteur de chaque instance étant par ailleurs appris en même temps que la tâche de classification. L'utilisation des traits de coréférence permet une amélioration d'environ 1% d'exactitude dans un système multiclasse au niveau 2, amélioration qui n'est probablement pas significative. Malgré ces résultats plutôt négatifs, il est clair qu'il existe un lien entre relations rhétoriques et relations de coréférence. La difficulté est donc de savoir comment les utiliser et comment mitiger les erreurs provenant des systèmes automatiques de coréférence.

Enfin, notons que différentes études ont proposé d'introduire de nouvelles informations, par exemple BIRAN et MCKEOWN (2013) utilisent un lexique d'affect (*Dictionnary of Affect in Language* (WHISSELL, 1989) [16]) et des scores de similarité cosinus entre les arguments, et WANG et al. (2010) utilisent un outil étiquetant automatiquement l'ordre temporel des évènements représentés par les verbes principaux de chaque argument, mais ces études ne permettent pas vraiment de conclure sur l'importance de ces informations bien qu'elles puissent sembler pertinentes. Une autre proposition relativement récurrente dans la littérature concerne la généralisation des informations lexicales à partir de méthodes de stemmatisation, de lemmatisation ou en se limitant aux lemmes de catégorie ouverte (MARCU et ECHIHABI, 2002 ; SPORLEDER, 2008), ou en se fondant sur Wordnet pour généraliser aux hyperonymes (SPORLEDER, 2008), éventuellement seulement pour les mots les moins fréquents, ou pour évaluer une similarité entre les arguments (en se fondant sur les liens de synonymie, antonymie, hyperonymie et hyponymie), mesure cependant moins efficace qu'une mesure cosinus ne nécessitant pas de ressources externes (BIRAN et MCKEOWN, 2013). Il a été également montré que se restreindre aux mots de catégorie ouverte entraîne une baisse de performance, probablement en partie à cause de la suppression des auxiliaires, ce qui limite l'information temporelle, et des modaux. Les autres méthodes ne se sont pas non plus montrées réellement efficaces. Concernant Wordnet, qui semble intuitivement une ressource pertinente, il serait intéressant de tester différentes niveaux de hiérarchie afin d'évaluer s'il existe un niveau qui conduit à une généralisation tout en permettant de conserver des distinctions pertinentes, même si l'utilisation de représentations de mots, par exemple clusterisées, pourrait permettre d'obtenir, selon la façon dont elles sont construites, une généralisation similaire sans utiliser une ressource construite manuellement.

16. Dans cette ressource, chaque mot est associé à un score selon trois dimensions (caractère plaisant, actif ou passif et facile ou difficile à imaginer). Le trait construit correspond à la moyenne de ces scores pour chaque argument.

Nous concluons cette section sur l'étude de Li et Nenkova (2014a) spécifiquement dédiée au problème de déséquilibre des classes. Ces auteurs testent différentes stratégies en binaire et en multiclasse pour le niveau 1, en ajoutant cependant une classe EntRel/NoRel ce qui, ajouté au fait qu'ils n'utilisent que les traits de type **Règles de production**, rend difficile une comparaison avec l'état de l'art. De plus, la comparaison entre les différentes stratégies ne nous semble ni très claire ni très juste. Ainsi, ils opposent d'abord un classifieur multiclasse, dans lequel est conservé le déséquilibre des classes, et un ensemble composé de quatre classifieurs binaires de type SVM, chacun sous-échantillonné, qui prédit la classe dont la distance à l'hyper-plan est la plus haute, ou la classe EntRel/NoRel si aucune distance n'est supérieure à 0. Ils concluent sur la supériorité de l'ensemble de classifieurs binaires mais n'indiquent pas pourquoi ils ne construisent pas également un classifieur binaire pour EntRel/NoRel et n'explicitent pas la stratégie implémentée par l'algorithme multiclasse (*one-vs-one* ou *one-vs-all*). Ils testent ensuite des stratégies permettant de ne pas supprimer d'exemples : sur-échantillonnage, et pondération des exemples au niveau de la fonction objective de l'algorithme. En binaire, les deux méthodes aboutissent à des résultats similaires et améliorent légèrement les performances obtenues avec le sous-échantillonnage. Le sous-échantillonnage restant plus performant en multiclasse, ils proposent de calculer les valeurs des traits sur l'ensemble des données, valeurs qui sont ensuite utilisées par l'ensemble de classifieurs binaires avec sous-échantillonnage, un pré-traitement assez lourd. Cette méthode conduit aux meilleures performances avec l'ensemble de classifieurs mais pas avec les modèles binaires séparés profitant plus de la pondération. Enfin, ils évaluent, mais en binaire uniquement, des combinaisons des classifieurs obtenus avec les différentes stratégies en mettant également en place un principe de vote. Il concluent finalement sur l'efficacité du sous-échantillonnage et la supériorité de la combinaison des classifieurs entraînés avec différentes stratégies, ce qui est cependant assez coûteux. Concernant le sous-échantillonnage, notons que Li et Nenkova (2014a) n'optimisent pas le sous-ensemble d'exemples à conserver (Park et Cardie, 2012), il n'est donc pas certain que l'on puisse reproduire leurs résultats. Quant à la comparaison entre un classifieur multiclasse et un ensemble de classifieurs binaires, elle n'est pas très juste puisque le premier ne repose sur aucune stratégie pour gérer le déséquilibre des classes. La dernière étude en date sur cette tâche (Rutherford et Xue, 2015), qui propose également des résultats en multiclasse au niveau 1, met en place une stratégie de pondération des instances au niveau de l'algorithme, stratégie à la fois plus simple, moins coûteuse (on ne duplique pas d'exemples, on n'optimise pas un ensemble d'entraînement) et finalement plus efficace.

Nous avons notamment présenté dans cette section les différents types d'informations utilisées pour représenter les données implicites, la façon dont elles étaient modélisées dans le cadre d'un système d'apprentissage statistique et les résultats obtenus dans le cadre de stratégies entièrement supervisées. Nous avons également vu que le problème au niveau 1 était généralement modélisé à travers plusieurs classifieurs binaires au lieu d'un classifieur multiclasse, configuration peu naturelle dans une configuration réaliste, et que différents types d'algorithmes avaient été utilisés, bien que le fait, notamment, de ne pas optimiser les hyper-paramètres ne permet pas de conclure véritablement sur la supériorité de l'un ou l'autre. Si les premières études ont généralement prôné l'utilisation de l'algorithme Naïf Bayes (Pitler et al., 2009 ; Park et Cardie, 2012 ; Rutherford et Xue, 2014), les suivantes ont plutôt mis en place des algorithmes de type régression logistique (Lin et al., 2009 ; Biran et McKeown, 2013 ; Rutherford et Xue, 2015) ou SVM (Ji et Eisenstein, 2014b). La modélisation des données repose sur un nombre important de ressources construites à la main qui ne sont disponibles que pour peu de langues. Notons de plus le risque avec ces représentations et tous les pré-traitements qui leur sont appliqués de construire des modèles sur-entraînés sur les données correspondant au domaine spécifique représenté par le PDTB, les articles journalistiques. Ce risque nécessitera une forme d'adaptation des modèles pour pouvoir les appliquer à d'autres types de données notamment pour construire des applications. Les études que

nous allons présenter dans la section suivante prennent le problème d'un point de vue différent en cherchant à améliorer la généralisation des modèles grâce à un apport massif de nouvelles données ou grâce à une transformation d'une représentation surfacique fondée sur des ressources acquises automatiquement.

3.3.4 Stratégies fondées sur une forme de non supervision

3.3.4.1 Utilisation des données explicites

La première étude qui s'est intéressée au problème des relations implicites (MARCU et ECHIHABI, 2002) a proposé une solution fondée sur une forme de non supervision. Plutôt que de se reposer sur un apport massif de connaissances (idéalement on devrait disposer d'un analyseur sémantique) les auteurs proposent d'augmenter le nombre d'exemples disponibles en annotant automatiquement de nouvelles données. Comme il est difficile d'annoter automatiquement des données implicites, ils utilisent le connecteur comme étiquette de la relation : ils extraient à partir de données brutes des exemples explicites, suppriment le connecteur, caractéristique des explicites, et obtiennent ainsi ce que nous appelons des *données (implicites) artificielles*, des implicites formés à partir d'explicites, par opposition aux *données (implicites) naturelles* annotées manuellement et qui sont de vrais implicites. En général, les études choisissent un sous-ensemble de connecteurs considérés comme non ambigus, en relation et en emploi, afin de limiter le bruit dans l'étiquetage.

RUTHERFORD et XUE (2015) décrivent cette stratégie comme s'inscrivant dans le cadre plus général de la supervision dite « distante »[17], parce qu'il s'agit « d'utiliser des connaissances ou des heuristiques pour obtenir à bas coût des *données faiblement étiquetées* » (nous traduisons). Les données sont *faiblement étiquetées* au sens où l'étiquette est peu sûre puisqu'elle est acquise automatiquement généralement grâce à des heuristiques. L'idée d'utiliser les données explicites pour identifier les données implicites est attrayante puisqu'elle permet d'obtenir relativement facilement de grande masse de données et, généralement, ajouter des données améliore les performances d'un système statistique. Cependant cette stratégie repose sur des hypothèses fortes[18] : il faut qu'il reste suffisamment d'informations, une fois le connecteur supprimé, pour pouvoir identifier la relation (*hypothèse de redondance du connecteur*) et il faut que les exemples explicites ressemblent suffisamment aux exemples implicites pour permettre une généralisation du modèle des premiers aux seconds (*hypothèse de similarité*). Nous détaillerons ces hypothèses dans le chapitre suivant en présentant des expériences situées dans ce même cadre.

L'étude de MARCU et ECHIHABI (2002) a donné lieu à une série d'études cherchant à utiliser au mieux ces données artificielles. Les premières études ont posé les bases de la méthode et identifié les problèmes qu'elle soulève (MARCU et ECHIHABI, 2002 ; SPORLEDER et LASCARIDES, 2005a ; BLAIR-GOLDENSOHN et al., 2007 ; SPORLEDER et LASCARIDES, 2008). Ces études ne donnent pas de scores sur le PDTB, du moins pas des scores comparables aux études présentées dans la section précédente. Les études suivantes, toutes sur le PDTB, ont proposé des stratégies pour intégrer de façon pertinente les données explicites soit en les sélectionnant (WANG et al., 2012 ; HONG et al., 2012 ; RUTHERFORD et XUE, 2015), soit en construisant de nouveaux traits à partir de ces données (ZHOU et al., 2010 ; BIRAN et MCKEOWN, 2013), soit en utilisant un algorithme adapté à cette configuration (LAN et al., 2013).

17. MARCU et ECHIHABI (2002) décrivent plutôt leur configuration comme un cadre non supervisé, mais les systèmes ne se fondent pas sur des données non annotées mais sur des données annotées automatiquement, nous conservons donc plutôt la terminologie de supervision distante.
18. Ces hypothèses ont été formulées d'abord par (SPORLEDER et LASCARIDES, 2008).

L'intuition qui sous-tend l'étude de MARCU et ECHIHABI (2002) est que l'inférence des relations discursives peut se faire sur la présence de paires de mots, comme « *pousser, tomber* » ou, pour l'exemple d'opposition qu'ils citent, « *embargo, legally* », et que ces paires de mots peuvent intervenir dans des données de type explicite et implicite. Ces oppositions ne peuvent pas forcément être récupérées grâce à des sources de connaissances type WordNet car elles ne se fondent pas sur le même type de relations sémantiques. Le problème de ces paires de mots c'est qu'elles sont relativement rares, et un modèle a besoin de les rencontrer suffisamment souvent pour construire une bonne estimation des paramètres associés. Or les données naturelles implicites sont peu nombreuses [19] : c'est pour remédier à ce manque de données dans le cadre de cette modélisation fondée sur les paires de mots que les auteurs proposent d'utiliser des exemples explicite extraits automatiquement à partir de données brutes (aucun corpus annoté manuellement en connecteurs n'existait à cette époque). L'annotation automatique se fait à partir de motifs (utilisant la position du connecteur et la ponctuation) fondés sur certains connecteurs considérés comme peu ou pas ambigus dans deux corpus bruts (corpus *Raw* [20] et corpus *Bllip* [21]). Ils s'intéressent à quatre relations du cadre de la RST : *Contrast*, *Explanation-evidence*, *Condition* et *Elaboration*. Ils ajoutent deux étiquettes dénotant l'absence de relation, l'un pour des segments appartenant à un même texte, l'autre pour des segments appartenant à des textes différents. Les arguments des connecteurs sont extraits par heuristique en se fondant sur des hypothèses simplificatrices : un argument recouvre au plus une phrase et on a au plus deux arguments par phrase. Le connecteur est enlevé des exemples extraits afin que le classifieur ne se fonde pas sur cet indice peu ambigu. Ils obtiennent finalement entre $900\ 000$ et 4 millions d'exemples artificiels par relation. Les auteurs rapportent notamment qu'un classifieur entraîné et testé sur ce type de données obtient des performances supérieures à la chance ce qui montre que le connecteur est redondant avec son contexte dans au moins une partie des données. Ils obtiennent ainsi $49,7\%$ d'exactitude pour 6 classes équilibrées. Les tests sur le corpus du RST DT ne sont par contre pas très révélateurs, d'abord parce qu'on ne sait pas exactement comment les connecteurs ont été ajoutés à l'annotation, ensuite parce que l'ensemble d'exemples d'évaluation comporte des exemples explicites et que, dans ce cas, les connecteurs sont conservés dans les données artificielles.

SPORLEDER et LASCARIDES (2005a) mènent une étude similaire mais utilisent des traits linguistiquement motivés plutôt que des paires de mots : la position des arguments (inter- ou intra-phrastiques), leur longueur, des informations morpho-syntaxiques ou lexicales (comme le chevauchement en mots, les lemmes des verbes et leurs classes Wordnet la plus haute). Une autre différence réside dans la taille plus restreinte du corpus artificiel. Constitué à partir de trois corpus (le *British National Corpus* [22], le *North American News Text Corpus* [23] et le *English Gigaword* [24]), il contient entre $2\ 000$ et $50\ 000$ exemples par relation. Leur étude se fonde de plus sur des relations de l'inventaire de la SDRT : *Contrast*, *Result*, *Explanation*, *Continuation* et *Summary*, relations choisies parce qu'elles peuvent apparaître avec un connecteur — donc on peut extraire des données artificielles pour ces relations — mais aussi sans connecteur — donc elles correspondent à des exemples pour lesquels on voudrait améliorer les performances. L'extraction des données se fait à partir de 55 connecteurs considérés comme non ambigus et une heuristique identifiant les arguments, inter- ou intra-phrastiques, à partir de la ponctuation et de la position du connecteur. Comme MARCU et ECHIHABI (2002), les auteurs testent un modèle entraîné et évalué sur des données artificielles équilibrées. Ils obtiennent $57,55\%$ d'exactitude pour 5 classes, reflétant encore une fois la re-

19. Les données explicites ne sont pas forcément plus nombreuses, mais elles contiennent un connecteur, un indice fort rend, à première vue du moins, moins nécessaire une modélisation riche.
20. Ce corpus est décrit par MARCU et ECHIHABI (2002) comme une concaténation de divers corpus non annotés anglais rendus disponibles par le *Linguistic Data Consortium*, il contient environ 1 milliard de mots.
21. Ce corpus contient près de 2 millions de phrases et est annoté automatiquement en syntaxe, il s'agit ici probablement d'une version plus ancienne du corpus que nous utilisons `https://catalog.ldc.upenn.edu/LDC2000T43`.
22. Corpus de 100 millions de mots, `http://www.natcorp.ox.ac.uk/`
23. Corpus de 350 millions de mots, `https://catalog.ldc.upenn.edu/LDC95T21`
24. Corpus de $1,7$ millions de mots, probablement la version de 2003 `https://catalog.ldc.upenn.edu/LDC2003T05`

dondance du connecteur. De plus, ce système est de plus de 20% meilleur qu'un système fondé sur les paires de mots, qui cependant ne profite pas ici de l'apport d'un aussi grand nombre de données. Les auteurs rapportent également des scores pour cette même tâche mais effectuée par des sujets humains. Ils obtiennent $71,25$% d'exactitude avec un score inter-annotateur de $0,61$ ce qui démontre la possibilité de retrouver la relation dans la plupart des cas.

BLAIR-GOLDENSOHN et al. (2007) adoptent aussi l'approche de MARCU et ECHIHABI (2002) et explorent l'une des voies de recherche proposées par ces auteurs : développer des méthodes générant moins de bruit pour collecter les données artificielles (segmentation en topiques et utilisation de la syntaxe). L'autre intérêt de cette étude est l'utilisation du PDTB notamment comme corpus de test (exemples implicites) mais aussi de manière informelle comme corpus artificiel (exemples explicites) cependant également annoté de manière automatique [25]. Nous ne sommes pas sûre que les auteurs utilisent les connecteurs annotés, mais ce corpus artificiel a au moins l'avantage d'être du même domaine que les données implicites, il est par contre probablement de taille plus restreinte, le PTB étant moins large que le GigaWord sur lequel est construit l'autre corpus artificiel. Les auteurs ne donnent cependant que peu de résultats sur cette expérience et se limitent à 3 classes (*Cause* [26], *Contrast* et absence de relation [27]). Comme nous l'avons déjà noté, ils équilibrent les corpus d'entraînement et de test, comme les études précédentes, ce qui rend leur configuration peu réaliste. Tous les modèles binaires obtiennent des résultats largement supérieurs à la chance. L'apport des méthodes permettant normalement d'obtenir un corpus moins bruité n'est pas évident. Lorsque l'évaluation est effectuée sur les données implicites du PDTB (version 1), donc des données naturelles, on observe une chute des performances, conclusion que l'on retrouve plus en détail dans (SPORLEDER et LASCARIDES, 2008).

L'étude de SPORLEDER et LASCARIDES (2008) a apporté une meilleure compréhension du problème posé par cette configuration à travers une étude extensive des données et des implications de la méthode. Ces auteurs reprennent une configuration similaire à celle présentée dans (SPORLEDER et LASCARIDES, 2005a) mais se testent cette fois sur des données non explicites (implicites mais aussi contenant un connecteur ambigu) issues du corpus RST DT étiquetées automatiquement avec les relations SDRT et vérifiées manuellement. Ils extraient entre $8\,000$ et 6 millions d'exemples artificiels par relation et disposent de $1\,051$ exemples non explicites manuellement annotés avec des classes relativement équilibrées (environ 250 exemples) sauf pour *Summary* (seulement 44 exemples). On a ici un écart important entre le nombre de données naturelles et artificielles, on espère donc compenser le manque d'exemples naturels par cet apport massif de nouvelles données. Ils rapportent comme précédemment des performances hautes pour un modèle entraîné et testé sur des données artificielles (60% d'exactitude avec des données relativement équilibrées, entre $8\,000$ et $16\,000$ exemples par relation). Par contre, ils constatent qu'un modèle entraîné sur des données naturelles conduit à des performances bien plus hautes ($40,30$% d'exactitude) que celles obtenues avec un modèle entraîné sur des données artificielles pourtant bien plus nombreuses ($25,80$%, donc une perte de près de 15%), et ceci indépendamment du classifieur (Naïf Bayes ou BoosTexter (SCHAPIRE et SINGER, 2000)) et du jeu de traits (paires de mots ou traits linguistiquement motivés essentiellement repris de (SPORLEDER et LASCARIDES, 2005a)). Notons que les performances obtenues avec les données artificielles restent cependant supérieures au 20% estimé pour un classifieur aléatoire. L'étude menée permet de conclure que les hypothèses faites par la méthode sont trop fortes. Le connecteur n'est pas toujours redondant/supprimable, sa

25. Le nombre d'exemples pour ce corpus n'est pas donné. Il est précisé que les auteurs utilisent la « version initiale » du corpus donc, probablement, le PDTB 1.0 (MILTSAKAKI et al., 2004) qui ne contenait pas l'annotation en sens des exemples explicites (PRASAD et al., 2008a).

26. *Cause* et *Consequence* dans le PDTB1.0, la première version de ce corpus contenant un jeu restreint de 7 relations différent de celui de sa version actuelle.

27. Ces exemples sont extraits automatiquement également pour l'ensemble de test sur le PDTB : les données correspondent à deux phrases séparées par au moins trois phrases.

suppression peut avoir un effet sur le discours, modifiant par exemple la relation inférable ce qui est évidemment préjudiciable dans le cadre d'un système d'apprentissage automatique. Les données ne sont pas similaires, sinon l'apport des données artificielles serait bénéfique au système : bien sûr, le bruit dans l'annotation automatique peut expliquer la baisse de performance, mais ce n'est probablement qu'un aspect du problème. Ces conclusions montrent que les données artificielles ne peuvent pas être utilisées directement comme données d'entraînement sans aucune adaptation.

La première étude à avoir cherché à gérer cette difficulté est celle présentée dans (WANG et al., 2012), leur solution consiste en une première phase de sélection des données avant l'entraînement des modèles. Plus précisément, ces auteurs proposent de remplacer l'opposition entre données naturelles et artificielles par une opposition entre données « typiques » et « non typiques » d'une relation. Ils utilisent une méthode de type clustering pour identifier les exemples typiques d'une relation parmi l'ensemble des données, naturelles et artificielles. Ces exemples typiques constituent ensuite les données d'entraînement de classifieurs, les autres exemples étant simplement supprimés. Pour déterminer le caractère typique d'un exemple, les auteurs partent d'un ensemble de règles manuelles (par exemple, la présence de « *first* » dans un argument pour *Temporal* ou la présence d'une paire de mots à polarité opposée pour *Comparison*) qui permettent de constituer un premier ensemble d'exemples graines, ensuite étendu aux exemples les plus similaires. Les auteurs se fondent sur l'article de YAROWSKY (1995), l'une des premières descriptions d'auto-entraînement (*self-training*). On peut voir le processus de sélection sous cet angle en considérant l'étiquetage des données selon les dimensions typiques ou atypiques, à la seule différence que l'on n'aboutit pas finalement à un classifieur à utiliser sur des données d'évaluation mais à un ensemble d'entraînement à utiliser pour un nouveau classifieur. Les auteurs considèrent que leur définition de la typicité d'un exemple doit permettre de constituer un ensemble d'entraînement plus représentatif et plus séparable, et que des exemples typiques peuvent être trouvés dans les deux ensembles de données. La sélection est par ailleurs une stratégie commune en adaptation de domaine sauf qu'appliquer une sélection quelque soit le domaine, ici au sens naturel *vs* artificiel, dont est issu l'exemple va à l'encontre du principe des méthodes d'adaptation où l'on conserve généralement tous les exemples cibles (ici naturels), parfois même en leur accordant plus de poids.

WANG et al. (2012) utilisent les données explicites du PDTB comme données artificielles, on n'a donc pas d'éventuelles différences en termes de domaine, au sens du genre journalistique ou de l'époque d'écriture des textes, ni de bruit mais un apport qui n'est pas réellement massif. Leur stratégie de sélection aboutit à un ensemble d'entraînement contenant $2\,494$ exemples pour *Temporal*, $3\,135$ pour *Contingency*, $2\,145$ pour *Comparison*, et $6\,795$ pour *Expansion*. On a donc en général plus, mais pas beaucoup plus, d'exemples que dans les données implicites originelles, légèrement moins pour *Contingency*. On note que pour *Temporal* et *Comparison*, le système conserve beaucoup plus d'exemples artificiels que d'exemples naturels (c'est le contraire pour *Expansion*, les proportions sont similaires pour *Contingency*). L'ensemble d'entraînement final contient $46,3\%$ d'exemples artificiels. Après la sélection, quatre classifieurs binaires et un classifieur multiclasse sont entraînés pour le niveau 1 de relation en utilisant une modélisation reprise globalement des études existantes. Leur système ne parvient pas cependant à surpasser les résultats de PARK et CARDIE (2012) en binaire avec des scores de F_1 de $14,7\%$ pour *Temporal*, de $48,5\%$ pour *Contingency*, de $28,5\%$ pour *Comparison* et de $71,1\%$ pour *Expansion*. En multiclasse, les auteurs rapportent un score d'exactitude de $72,2\%$ à notre connaissance jamais égalé ni surpassé et, par ailleurs, rarement cité. On n'a cependant aucune idée des scores par classe obtenus avec ce classifieur. Si les scores obtenus, du moins en binaire, ne sont pas meilleurs que dans les études précédentes, cette étude démontre au moins la possibilité d'utiliser les données artificielles sans observer une dégradation importante des performances. L'un des problèmes de cette méthode est le fait d'être fondée sur des règles manuelles qu'il faut donc définir et qui reposent sur l'utilisation de ressources comme

les lexiques de polarité. Un autre problème vient du fait de sélectionner, et notamment parmi les données implicites, et donc de supprimer des exemples, de réduire les données disponibles et, éventuellement, de rater des caractéristiques rares mais cruciales. Les auteurs précisent que $61,8\%$ des exemples d'évaluation sont considérés comme typiques, donc le système semble quand même se généraliser à certains exemples atypiques. Ceci peut s'expliquer par le fait que cette seconde phase n'est pas fondée sur des listes de règles de décision mais sur un modèle capable d'effectuer des corrélations plus fines entre les traits. Il serait intéressant de savoir en quoi ces presque 40% d'exemples sont atypiques.

Des stratégies reposant sur une sélection ont également été explorées dans HONG et al. (2012) et RUTHERFORD et XUE (2015). HONG et al. (2012) proposent d'extraire des exemples artificiels en se fondant sur leur similarité avec les exemples implicites disponibles, la similarité étant calculée à partir d'une mesure cosinus sur des bigrammes. L'annotation du sens de l'exemple artificiel se fait également en utilisant une similarité, cette fois entre cet exemple et des exemples explicites mettant en jeu un connecteur non ambigu (un connecteur est non ambigu s'il exprime le même sens dans 90% des cas dans le PDTB). Le système final identifie une relation implicite avec un système de vote parmi les n exemples artificiels les plus similaires. Les auteurs ne donnent que des scores d'exactitude pour les systèmes binaires ce qui, ajouté au fait qu'ils s'évaluent sur des sections différentes de celles utilisées généralement, rend difficile l'estimation des performances du système. RUTHERFORD et XUE (2015) exploitent également des mesures de similarité entre exemples naturels et artificiels pour effectuer une sélection parmi le second ensemble. Plus précisément, ils définissent deux critères pour identifier les exemples artificiels pertinents par rapport à la tâche, critères par ailleurs liés aux hypothèses faites par la méthode, et qui utilisent le fait que l'on dispose de connecteurs annotés pour les données implicites dans le PDTB :

— le *taux d'omission* permet d'identifier les connecteurs redondants, il correspond à la fréquence du connecteur dans les exemples implicites par rapport au nombre total d'occurrences du connecteur,

— la *différence de contexte* permet d'identifier les exemples artificiels les plus similaires aux implicites, c'est-à-dire les cas où la suppression du connecteur ne change pas la relation inférée. Cette mesure correspond à la divergence Jensen-Shannon entre la distribution des implicites pour un connecteur et celle des explicites pour ce même connecteur, une distribution correspondant à un modèle unigramme sur l'ensemble des mots des deux arguments.

Les auteurs utilisent 54 connecteurs qui apparaissent dans les données implicites et explicites et qui sont considérés comme assez peu ambigus (sens le plus fréquent dans 90% des données). Les scores sont calculés à partir des données implicites et explicites du PDTB. Ils utilisent un algorithme de clustering hiérarchique pour identifier des ensembles de connecteurs selon les valeurs obtenues pour les deux critères. Parmi les quatre classes de connecteurs obtenues, ils distinguent en particulier une classe de connecteurs « librement supprimables » (taux d'omission haut et divergence faible). Les données artificielles sont extraites du PDTB et d'un corpus brut, le GigaWord, à partir de motifs fondés sur la catégorie morpho-syntaxique du connecteur, sa position et la ponctuation. L'heuristique pour les arguments permet d'extraire des exemples dont la configuration ressemble à celle des implicites dans le PDTB, donc inter-phrastiques [28]. Finalement ils entraînent un classifieur multiclasse de type régression logistique au niveau 1 avec les traits issus de (PITLER et al., 2009) et un schéma de pondération d'instances afin de gérer le déséquilibre en classe, plus crucial encore de par l'ajout des exemples artificiels qui faussent la distribution naturelle des données. Ils rapportent en multiclasse un score d'exactitude de $57,1\%$ et un score de macro-F_1 de $40,5\%$, contre respectivement $55,0\%$ et $38,4\%$ sans l'ajout de données artificielles. Ils montrent que les données

28. Pour les conjonctions de subordination, ils ne conservent que les connecteurs intra-phrastiques. Pour les autres catégories, ils ne conservent que les cas où le connecteur débute une phrase et est suivi d'une virgule, le premier argument étant alors la phrase précédente et le second, la phrase contenant le connecteur.

artificielles correspondant à des connecteurs librement supprimables conduisent aux meilleures performances, performances qui semblent pouvoir encore être améliorées par l'ajout de nouvelles données (au maximum ici $200\ 000$). Ils reproduisent aussi la configuration binaire et rapportent des scores de F_1 de $33,3\%$ pour *Temporal*, $53,8\%$ pour *Contingency*, $41,0\%$ pour *Comparison*, et $69,4\%$ pour *Expansion*, améliorant donc les scores précédemment cités. Leur stratégie de sélection, en plus de permettre une amélioration importante des scores, est particulièrement attrayante car elle est liée aux hypothèses faites par la méthode. Elle nécessite cependant l'annotation de connecteurs implicites, une information qui ne sera pas forcément disponible pour toutes les langues et domaines. Par ailleurs, il serait intéressant de savoir si l'utilisation de méthode d'adaptation de domaine au niveau des classifieurs finaux ne permettrait pas des améliorations supplémentaires.

En dehors des méthodes de sélection, il a également été proposé d'utiliser les données artificielles dans le cadre de l'apprentissage multi-tâche (LAN et al., 2013), cadre qui permet d'apprendre une tâche principale en même temps que d'autres tâches auxiliaires qui lui sont liées. Les tâches sont apprises de manière jointe et l'algorithme peut profiter des points communs entre les tâches pour s'améliorer sur la tâche principale. C'est un cas d'apprentissage par transfert (PAN et YANG, 2010), également utilisé en adaptation de domaine. LAN et al. (2013) utilisent l'algorithme multi-tâche ASO (*Alternating Structure Optimization*) (ANDO et ZHANG, 2005). La tâche d'identification des relations implicites est la tâche principale et elle est informée par la tâche d'identification de la relation dans les données artificielles : on apprend ainsi en théorie ce qui est partagé entre les tâches sans introduire du bruit provenant des tâches auxiliaires. Les auteurs utilisent des données artificielles construites à partir des exemples explicites du PDTB ou extraites d'un corpus brut, ici le *Bllip* [29]. Pour le PDTB, au lieu d'utiliser le sens annoté, les relations correspondent au sens le plus fréquent pour le connecteur annoté. Ils n'expliquent pas par contre comment ils extraient les connecteurs et les arguments dans le *Bllip*. Ils obtiennent $26\ 412$ exemples à partir de ce corpus, ce qui est relativement peu, même pas le double du nombre d'exemples explicites dans le PDTB. LAN et al. (2013) utilisent moins de traits que dans les études précédentes (uniquement **Verbe**, **Modalité** et **Polarité**) et construisent un classifieur binaire pour chaque relation aux niveaux 1 et 2 [30]. Les auteurs rapportent des scores en utilisant uniquement les données d'un type, implicite ou artificiel construit sur le PDTB et/ou sur le *Bllip*, et des combinaisons (union) de ces données. Comme précédemment, on observe une chute des performances pour les modèles entraînés sur des données artificielles mais on note que les données *Bllip* ne sont pas toujours moins bonnes que les données PDTB. L'apprentissage multi-tâche permet d'améliorer les performances pour tous les classifieurs avec finalement des scores de F_1 de $29,51\%$ pour *Temporal*, $47,52\%$ pour *Contingency*, $31,53\%$ pour *Comparison*, et $70,01\%$ pour *Expansion*. Les performances pour *Temporal* n'ont été surpassées que très récemment (RUTHERFORD et XUE, 2015). Les résultats au niveau 2 ne sont pas comparables à ceux de (LIN et al., 2009) puisqu'ici les auteurs utilisent des classifieurs binaires et un ensemble d'évaluation différent. Pour ce niveau, la méthode permet d'obtenir des améliorations par rapport à un simple entraînement sur les données implicites disponibles, améliorations cependant assez faibles (gains en F_1 entre $0,5$ et $2,5\%$), et quatre relations reçoivent un score de F_1 nul dont *List* qui correspondait à une F_1 de 23% dans (LIN et al., 2009), ceci étant peut-être dû à la modélisation limitée des données.

BIRAN et MCKEOWN (2013) envisagent le problème d'une manière différente : ils utilisent les connecteurs pour construire un nouveau trait qui associe à une instance implicite sa similarité par rapport aux connecteurs. Ils extraient les paires de mots (ou plutôt de stems, afin d'augmenter la généralisation) apparaissant autour d'un connecteur au sein d'une phrase, ce qui est une simplification de la segmentation de arguments et introduit probablement du bruit notamment pour

29. `https://catalog.ldc.upenn.edu/LDC2008T13`
30. Pour le second niveau LAN et al. (2013) enlèvent six relations, et non cinq comme LIN et al. (2009), probablement parce que cette sixième relation correspond à peu d'exemples explicites (*Pragmatic cause*).

les adverbiaux et les connecteurs discontinus. Leur stratégie consiste à considérer un connecteur comme un document contenant des termes, les paires de stems. Ils calculent un score permettant d'estimer la pertinence d'une instance implicite par rapport au connecteur/document, une méthode inspirée par le cadre de la classification de documents. Chaque terme associé à un connecteur reçoit un score correspondant au ratio entre la fréquence normalisée du terme (*TF*) ou la mesure d'information mutuelle pour un point (*Pointwise Mutual Information*) et l'inverse de la fréquence du document (*IDF*). La similarité entre une instance implicite et chaque connecteur est calculée à partir du cosinus entre chaque paire composant l'instance, représentée par sa fréquence, et les paires composant le document/connecteur, représentées par la mesure de TF-IDF ou de PMI-IDF. En utilisant cette seule similarité comme trait, les auteurs obtiennent des scores de F_1 de $19,54\%$ pour *Temporal*, $44,03\%$ pour *Contingency*, $24,38\%$ pour *Comparison*, et $66,48\%$ pour *Expansion*, surpassant donc (PITLER et al., 2009) pour toutes les relations sauf *Contingency*. En ajoutant d'autres traits repris des études précédentes et de nouveaux traits construits à partir de WordNet, d'informations de similarité, d'affect et de négation, ces performances augmentent avec au final des scores de F_1 de $20,23\%$ pour *Temporal*, $46,94\%$ pour *Contingency*, $25,4\%$ pour *Comparison*, et $75,87\%$ pour *Expansion*, améliorant le système de PARK et CARDIE (2012) uniquement pour *Contingency*.

Finalement, deux études ont proposé une stratégie différente, à mi-chemin entre l'utilisation des explicites comme données d'entraînement et la recherche de nouveaux indices pertinents. LAPATA et LASCARIDES (2004) et ZHOU et al. (2010) proposent d'envisager le problème de l'inférence d'une relation implicite sous l'angle de la prédiction d'un connecteur. Les premiers se focalisent sur les connecteurs temporels et cherchent uniquement à retrouver le connecteur et non la relation associée. Les seconds mènent des expériences sur le PDTB au niveau 1 de relation. Ils entraînent un modèle de langue sur des données brutes (3-grammes), ils combinent ensuite les arguments des exemples implicites avec chacun des connecteurs du PDTB et utilisent le modèle de langue pour calculer un score de perplexité pour les instances pseudo-explicites ainsi formées. Ils testent ensuite deux stratégies : soit les 60 meilleurs connecteurs, par ordre de perplexité croissante, sont ajoutés à un ensemble de traits issus d'études précédentes, soit ils utilisent seulement le connecteur prédit avec le meilleur score et assignent à l'exemple le sens le plus fréquemment déclenché par ce connecteur. Ce sens le plus fréquent est récupéré soit à partir des exemples explicites du PDTB soit à partir des exemples implicites pour lesquels, rappelons-le, les annotateurs devaient insérer (au moins) un connecteur. Leurs meilleurs résultats sont obtenus avec la première stratégie, l'ajout des connecteurs prédits comme traits, qui leur permet d'améliorer un système n'utilisant pas ces traits de 1 à 4 points en F_1 selon les relations. Ils obtiennent finalement au mieux des scores de F_1 de $20,30\%$ pour *Temporal*, de $47,16\%$ pour *Contingency*, de $31,79\%$ pour *Comparison* et de $70,11\%$ pour *Expansion* donc n'améliorant les résultats de PITLER et al. (2009) que pour *Comparison* et correspondant à des performances inférieures à PARK et CARDIE (2012). D'une certaine façon, en utilisant les données artificielles à l'entraînement, on essaye également de retrouver le connecteur ou plutôt un cluster de connecteurs correspondant à une relation.

3.3.4.2 Modification ou apprentissage de la représentation des données

Les dernières études pour l'identification des relations discursives suivent une tendance actuelle du TAL, à savoir l'utilisation de représentations sur les mots apprises de manière non supervisée. Comme la tâche consiste à classer des paires de segments, donc des paires de séquences de mots, les études proposent des solutions pour constituer un vecteur composite correspondant à la paire de segments à partir de représentations de mots.

RUTHERFORD et XUE (2014) reprennent la représentation des données sous forme de paires de mots et résolvent le problème d'éparpillement en remplaçant chaque mot de la paire par un code binaire correspondant au cluster auquel il appartient, clusters bien sûr largement moins nombreux (3 200) que les mots du vocabulaire (plusieurs millions). L'utilisation de ces codes permet également d'effectuer une généralisation de type sémantique et syntaxique sans pour autant faire appel à des ressources construites manuellement, la représentation étant obtenue en utilisant un algorithme de clustering hiérarchique (BROWN et al., 1992) sur un large corpus de données brutes en se fondant sur la co-occurence des mots dans une fenêtre. L'aspect hiérarchique de cet algorithme permet d'obtenir des classes de mots avec différents niveaux de granularité. La représentation utilisée est disponible librement pour l'anglais et peut être facilement induite pour toute langue disposant d'un ensemble de textes bruts. Cette approche est donc beaucoup plus généralisable que celles reposant sur un large éventail de ressources du type de (PITLER et al., 2009). Elle permet également de se passer de l'annotation automatique de données explicites : même si elle nécessite également un traitement sur un texte brut, à moins de disposer d'un clustering déjà construit, elle évite la phase de définition d'une heuristique pour identifier les connecteurs et leurs arguments.

RUTHERFORD et XUE (2014) utilisent, en plus des paires de codes de cluster, des traits repris de (PITLER et al., 2009) (**Premier, dernier, trois premiers mots**, **Modalité**, **Catégories sémantiques**, **Polarité**, **Nombre** et **Verbe**) ainsi que les règles de production, des informations de coréférence [31] et un trait qu'ils nomment « time expressions » mais qu'ils ne définissent pas. Ils présentent des résultats binaires pour les relations de niveau 1, et donnent des résultats en incluant ou non les EntRel dans la relation *Expansion*. Ils utilisent une stratégie de pondération d'instance pour gérer le déséquilibre des classes et optimisent un filtre en fréquence sur les traits. Ils obtiennent au mieux des scores de F_1 de $28,69\%$ pour *Temporal*, $54,42\%$ pour *Contingency*, $39,70\%$ pour *Comparison* et $80,44\%$ pour *Expansion* en incluant EntRel, $70,23\%$ sans inclure EntRel. Ces deux derniers scores montrent bien que les EntRel jouent un rôle crucial dans les performances obtenues pour la classe *Expansion*. Ce sont, à ce jour, les meilleurs scores pour cette tâche pour les relations *Contingency* et *Expansion*. Pour *Temporal* et *Comparison*, ces scores ont été dépassés dans leur étude suivante (RUTHERFORD et XUE, 2015) utilisant des données artificielles. Afin d'évaluer l'impact de différents traits, RUTHERFORD et XUE (2014) mènent une étude en supprimant les groupes de traits un par un. Ils montrent ainsi que la suppression des traits construits à partir des clusters Brown conduit aux baisses les plus importantes pour toutes les relations. Suivent ensuite les règles de production et les traits lexicaux sur les premiers et derniers mots des arguments. Cependant, ils ne comparent pas à l'utilisation de traits de type paires de mots sans utilisation des clusters Brown et ils ne donnent pas de résultats utilisant seulement les paires de mots et les paires de codes de cluster sans les autres traits. Nous verrons dans le chapitre 5 que la méthode qu'ils utilisent ne donne en fait pas de meilleurs résultats que l'utilisation des simples paires de mots. Les bonnes performances observées ici sont plutôt dues, selon nous, à l'utilisation d'une méthode plus fine pour gérer le déséquilibre des classes et à l'optimisation d'un filtre en fréquence qui remplace le seuil fixé *a priori* dans les études précédentes. Les auteurs mènent une étude intéressante sur les clusters, montrant que les regroupements qu'ils effectuent sont pertinents vis-à-vis des relations de discours. Ainsi pour *Comparison*, les paires les plus informatives mettent en lien des paires de mots appartenant aux mêmes clusters, ce qui correspond à l'idée d'une relation servant à mettre en parallèle ou en contraste des caractéristiques d'une même entité. Quelque part, ils agissent de la même façon que ce que l'on pouvait attendre des traits de coréférence. De même, pour *Temporal* les paires mettent souvent en jeu des expressions temporelles mais aussi des mots exprimant un

31. Ces traits correspondent au nombre de paires coréférentielles entre les deux arguments, à un trait binaire encodant la présence de noms similaires ou coréférentiels arguments d'un prédicat similaire, un trait binaire indiquant si les verbes principaux des deux arguments ont le même sujet ou pas et un trait binaire indiquant si les verbes principaux sont similaires. La similarité n'est pas définie, l'égalité des sujets correspond au fait qu'ils sont coréférentiels ou appartiennent au même cluster.

changement (« *increase, rose, loss* »). Concernant la relation *Contingency*, ils trouvent des paires de mots sémantiquement liés comme des termes techniques correspondant au commerce et aux échanges liés à des termes économiques.

JI et EISENSTEIN (2014a) proposent une approche se fondant sur un apprentissage d'une représentation pour les arguments à partir des arbres syntaxiques. Cette approche s'apparente donc à celle qu'ils mettent en place pour la construction d'un analyseur discursif sur le RST DT (JI et EISENSTEIN, 2014b). La méthode utilisée est cependant différente. Partant d'une représentation vectorielle pré-existante pour les mots (plongements lexicaux), ils calculent la représentation de chaque nœud non terminal en effectuant une combinaison des représentations de ses enfants, à la manière d'un modèle de neurones récursif utilisant la fonction de tangente hyperbolique. Ils obtiennent ainsi finalement à la racine de l'arbre un vecteur représentant l'argument. Cette représentation est augmentée pour tenir compte de lien de coréférence. Plus précisément, ils calculent de nouveaux vecteurs pour chaque paire de mentions coréférentes. Pour cela, ils effectuent un retour sur l'arbre en partant cette fois de sa racine et recalculent les vecteurs pour chaque constituant en combinant de la même manière que précédemment la représentation de son parent et de son voisin (les arbres étant binaires). Ils obtiennent ainsi des vecteurs pour chaque entité en lien de coréférence. La prédiction d'une relation se fait en sommant un premier terme correspondant au produit des vecteurs représentant chaque argument et une matrice de paramètres associés, et un second terme correspondant à une somme sur les vecteurs de coréférence d'un produit similaire mettant en jeu un autre vecteur de paramètres. Un troisième terme est ajouté pour prendre en compte d'autres types de traits. Tous ces paramètres ainsi que les deux paramètres correspondant aux compositions effectuées sur les arbres sont appris conjointement à la tâche de classification ce qui permet d'obtenir une représentation adaptée au problème. Les auteurs utilisent un système automatique pour annoter les entités et les liens de coréférence. En plus de ces traits vectoriels, ils ajoutent les traits de (LIN et al., 2009) et utilisent le même filtre fondé sur l'information mutuelle. Ils reproduisent le système de LIN et al. (2009), mais en utilisant un autre algorithme et se comparent à un système de référence où chaque argument est représenté par la somme des vecteurs représentant chaque mot. Ils donnent des résultats en multiclasse au niveau 2 et en binaire au niveau 1. Leur meilleur système pour le niveau 2 correspond à un score d'exactitude de $43,56\%$ soit supérieur de plus de 3 points au système de LIN et al. (2009). L'utilisation des traits de coréférence permet un gain d'environ 1 point en exactitude. Le système de référence correspondant à utiliser des représentations existantes n'obtient que $28,73\%$ d'exactitude. Pour les expériences en binaire, les auteurs incluent les EntRel dans *Expansion*. Ils obtiennent des scores de F_1 de $26,91\%$ pour *Temporal*, $51,39\%$ pour *Contingency*, $35,84\%$ pour *Comparison* et $79,91\%$ pour *Expansion*. Ils améliorent assez largement les scores de PARK et CARDIE (2012) pour *Comparison*, l'amélioration est moins forte pour les autres relations. Leurs scores sont inférieurs à ceux rapportés dans (RUTHERFORD et XUE, 2014) alors que la méthode mise en place est bien plus complexe.

3.3.5 Résumé des scores sur les études existantes

Pour le niveau 1, les scores par classifieur binaire sont reproduits dans le tableau 3.2. Notons que toutes ces études ne sont pas forcément directement comparables.

En multiclasse au niveau 1, RUTHERFORD et XUE (2015), la dernière étude en date, rapportent $40,5\%$ de macro-F_1 et $57,1\%$ d'exactitude. Nous reproduisons les scores par classe dans la table 3.3a. Pour le niveau 2, LIN et al. (2009) obtiennent $40,2\%$ d'exactitude et JI et EISENSTEIN (2014a) rapportent $43,56\%$. Ces derniers ne donnent cependant pas de scores par relation. Nous reproduisons les scores par relation de LIN et al. (2009) dans la table 3.3b.

Étude	*Temporal*	*Contingency*	*Comparison*	*Expansion*
(Pitler et al., 2009)	21, 88	47, 13	21, 88	76, 42
(Zhou et al., 2010)	20, 30	47, 16	31, 79	70, 11
(Park et Cardie, 2012)	26, 57	49, 82	31, 32	79, 22
(Wang et al., 2012)	14, 7	48, 5	28, 5	71, 1
(Lan et al., 2013)	29, 51	47, 52	31, 53	70, 01
(Biran et McKeown, 2013)	20, 23	46, 94	25, 4	75, 87
(Li et Nenkova, 2014a)	23, 88	48, 38	29, 89	57, 43
(Li et Nenkova, 2014b)	19, 97	48, 90	27, 78	-
(Rutherford et Xue, 2014)	28, 69	**54, 42**	39, 70	**80, 44** (70, 23)
(Ji et Eisenstein, 2014a)	26, 91	51, 39	35, 84	79, 91
(Rutherford et Xue, 2015)	**33, 3**	53, 8	**41, 0**	69, 4

Table 3.2.: Résumé des scores de F_1 obtenus dans les études existantes sur le PDTB pour le niveau 1 de relation. Pour (Rutherford et Xue, 2014), le nombre entre parenthèses correspond au score obtenu sans inclure les EntRel.

(a) Scores par classe (Rutherford et Xue, 2015).

Relation	P	R	F_1
Temporal	38, 5	9, 1	14, 7
Contingency	49, 3	39, 6	43, 9
Comparison	44, 9	27, 6	**34, 2**
Expansion	61, 4	78, 8	**69, 1**

(b) Scores par classe (Lin et al., 2009).

Relation	P	R	F_1
Asynchronous	50	8	13
Synchonous	0	0	0
Cause	39	76	**51**
Pragmatic Cause	0	0	0
Contrast	61	9	15
Concession	0	0	0
Conjunction	30	51	38
Instantiation	67	39	49
Restatement	48	27	35
Alternation	0	0	0
List	80	13	**23**

Table 3.3.: Scores de précision (« P »), rappel (« R ») et F_1 par relation pour le système état de l'art en multiclasse au niveau 1 (Rutherford et Xue, 2015) et le système multiclasse au niveau 2 présenté dans (Lin et al., 2009), la dernière étude au niveau 2 (Ji et Eisenstein, 2014a) ne rapportant pas de scores pas classe.

3.4 Systèmes de référence

Nous donnons dans cette section des résultats de référence pour différents jeux de traits sur le PDTB, corpus que nous utilisons principalement dans la suite de cette thèse et seul corpus pour lequel nous pouvons vraiment nous comparer à un état de l'art.

3.4.1 Configuration et nombre d'exemples disponibles

Comme nous l'avons vu précédemment, construire un système sur le PDTB requiert de faire des choix concernant certains points de la configuration. Notre configuration est donc la suivante :

— Multiclasse *vs* binaire : nous construisons des systèmes pour le niveau 1, en binaire et en multiclasse, et pour le niveau 2, en multiclasse uniquement. Rappelons que la configuration binaire est la plus répandue au niveau 1, mais qu'un système multiclasse est plus réaliste, c'est aussi la configuration choisie dans la dernière étude à ce niveau (Rutherford et Xue, 2015). Pour le niveau 2, la configuration multiclasse est la plus répandue ;

— Déséquilibre des classes : nous choisissons la stratégie fondée sur une pondération des instances utilisées dans la plupart des dernières études (Li et Nenkova, 2014a ; Rutherford et Xue, 2014 ; Rutherford et Xue, 2015) ;

— Sections d'entraînement et d'évaluation : nous suivons la tendance générale, à savoir que l'ensemble d'entraînement correspond aux sections 2-20, celui de développement aux sections 00,01,23 et 24 et celui de test aux sections 21 et 22. Cependant, nous utilisons ces mêmes sections pour le niveau 2, car se limiter à la section 23 comme le font LIN et al. (2009) pose problème pour des tests de significativité (seulement 766 exemples [32]) ;

— Annotations multiples : nous ne conservons que la première annotation car nous considérons qu'il n'est pas souhaitable dans la construction d'un système statistique d'avoir des exemples annotés différemment. Idéalement, il faudrait regarder ces annotations et choisir la plus informative, mais comme ce phénomène concerne moins de 2% des données, nous pensons que cela ne pose pas de problème majeur ;

— AltLex : nous n'incluons pas les AltLex (624 exemples au total) car nous considérons que ces exemples sont plutôt des exemples explicites puisque les annotateurs ont annoté dans le texte une forme déclenchant la relation, contrairement aux implicites pour lesquels aucune marque n'est identifiée ;

— EntRel : nous n'incluons pas les EntRel dans les données implicites de la relation d'*Expansion* car ces données nous semblent d'un autre type, plus proche de la tâche de coréférence et reposent sur des indices spécifiques. De plus, ces données sont assez nombreuses (5 210 exemples) ce qui a probablement une influence importante sur les scores, ce dont rendent compte les scores rapportés par RUTHERFORD et XUE (2014) avec un gain d'environ 10 points en F_1 quand ces exemples sont inclus. La tendance est par ailleurs désormais à les exclure (RUTHERFORD et XUE, 2014 ; RUTHERFORD et XUE, 2015).

— NoRel : nous ne faisons pas de classe séparée pour les NoRel car nous pensons qu'ils correspondent à une autre tâche, celle d'attachement, qui doit se faire séparément par un classifieur probablement fondé sur des indices différents. De plus, il nous semble que le PDTB se prête assez mal à ce problème puisqu'il y a très peu d'exemples de NoRel (254 exemples au total) et que le corpus ne correspond pas à une couverture totale. Il semble plutôt étrange que des phrases ne soient reliées à rien dans le cadre d'un corpus discursif.

Nous résumons les données disponibles pour le niveau 1 dans le tableau 3.4 et celles disponibles pour le niveau 2 dans le tableau 3.5. Pour le niveau 2, comme le montre le tableau, certaines relations (en gras) correspondent à trop peu d'exemples, entre 1 et 2 exemples dans l'ensemble du corpus. Ces relations ne sont donc pas prises en compte dans la construction des modèles, ce qui est le cas dans les études précédentes (LIN et al., 2009 ; JI et EISENSTEIN, 2014a).

Notons que le nombre total d'exemples n'est pas le même aux différents niveaux de relation. Ceci est dû à la possibilité pour les annotateurs de choisir le niveau de sens qu'ils veulent, certaines instances ne sont donc annotées qu'au premier niveau de relation. Pour constituer les ensembles d'exemples de niveau 1, on utilise toutes les données disponibles dans le corpus en transformant leur étiquette vers le niveau 1. Pour le niveau 2 par contre, on ignore les exemples qui ne sont annotés qu'au niveau 1 comme le font LIN et al. (2009).

3.4.2 Algorithmes de classification par régression logistique

Pour rappel, la tâche de classification correspond à l'assignation à un exemple d'une classe parmi un ensemble prédéfini. En apprentissage supervisé, on apprend la fonction de classification à partir des données d'entraînement, un ensemble d'instances étiquetées. Dans nos expériences, nous utilisons un modèle par régression logistique ou maximum d'entropie (BERGER et al., 1996) (le premier terme étant plutôt utilisé pour le cas binaire et le second en multiclasse) qui donne

32. LIN et al. (2009) indiquent 780 exemples mais ce chiffre inclut des annotations multiples, en tout il y a 769 exemples implicites dans la section 23 dont 3 ne sont pas annotés au niveau 2.

Relation	Entraînement	Développement	Test	Total (%)
Temporal	665	93	68	826 (5, 14%)
Contingency	3 281	628	276	4 185 (26, 07%)
Comparison	1 894	401	146	2 441 (15, 21%)
Expansion	6 792	1 253	556	8 601 (53, 58%)
Total	12 632	2 375	1 046	16 053

Table 3.4.: Copus PDTB : nombre d'exemples implicites par relation pour les ensembles d'entraînement, de développement et de test au niveau 1 de relation. Dans le cas de classifieurs binaires, tous les exemples des classes autres que la classe considérée sont utilisés comme données négatives à l'entraînement.

Relation	Entraînement	Développement	Test	Total (%)
Asynchronous	517	79	54	650 (5, 24%)
Synchronous	147	14	14	175 (1, 41%)
Cause	3 227	617	269	4 113 (33, 15%)
Pragmatic Cause	51	11	7	69 (0, 56%)
Condition	1	0	0	1
Pragmatic Condition	1	0	0	1
Contrast	1 566	369	128	2 063 (16, 63%)
Pragmatic Contrast	2	0	0	2
Concession	180	22	17	219 (1, 76%)
Pragmatic Concession	1	0	0	1
Conjunction	2 805	435	200	3 440 (27, 73%)
Instantiation	1 061	216	118	1 395 (11, 24%)
Restatement	2 376	521	211	3 108 (25, 05%)
Alternative	146	25	9	180 (1, 45%)
Exception	1	0	0	1
List	330	44	12	386 (3, 11%)
Total	12 412	2 353	1 039	15 804
Total ajusté	12 406	2 353	1 039	15 798 (100%)

Table 3.5.: Corpus PDTB : nombre d'exemples implicites pour les ensembles d'entraînement, de développement et de test au niveau 2 de relation. Les relations en gras ne sont pas prises en compte dans les expériences. Le pourcentage d'exemple dans la dernière colonne est calculé par rapport au total ajusté, donc sans prendre en compte les six relations en gras.

de bonnes performances pour différents problèmes de TAL. C'est un modèle probabiliste — on obtient une distribution de probabilité sur les classes pour chaque exemple —, et un modèle discriminant — on modélise directement la probabilité conditionnelle des classes sachant les données. Ce type de modèle a aussi l'avantage de permettre l'ajout de nombreux descripteurs potentiellement redondants sans faire d'hypothèses d'indépendance.

Pour décrire ce modèle, on note $\mathcal{X}$ l'ensemble des données en entrée et $\mathcal{Y}$ les sorties possibles. On veut apprendre une règle de prédiction $h : \mathcal{X} \rightarrow \mathcal{Y}$. Cette prédiction se fonde sur une représentation des données sous forme de vecteurs de traits, $\Phi(x) \in \mathbb{R}^d$ en binaire $\Phi(x, y) \in \mathbb{R}^d$ en multiclasse, et sur un vecteur de poids $\mathbf{w} \in \mathbb{R}^d$, les paramètres du modèle. À chaque trait ϕ_i est associé un poids w_i. Le vecteur de paramètres est appris à l'entraînement à partir d'un ensemble de données étiquetées $((x_1, y_1), ..., (x_n, y_n)) \subset \mathcal{X} \times \mathcal{Y}$.

Dans le cas binaire, il est plus simple de donner les valeurs 1 et -1 comme étiquettes de chacune des classes. Le principe de la régression logistique est de fournir en sortie la probabilité de l'étiquette, la fonction d'hypothèse h a donc ses valeurs dans l'intervalle $[0, 1]$. On cherche à modéliser $h(x) = P(y \in \mathcal{Y}|x \in \mathcal{X})$, ou dans le cas binaire, plus simplement $P(Y = 1|X = x)$, puisqu'alors $P(Y = -1|X = x) = 1 - P(Y = 1|X = x)$. Comme pour d'autres modèles linéaires, l'hypothèse se fonde sur le produit scalaire $\mathbf{w} \cdot \mathbf{\Phi}(\mathbf{x}, \mathbf{y})$ mais on utilise la fonction dite *logit* pour s'assurer des valeurs $h(x)$ dans le bon intervalle :

$$h(x) = \frac{e^{\mathbf{w}\cdot\mathbf{\Phi}(\mathbf{x})}}{1 + e^{\mathbf{w}\cdot\mathbf{\Phi}(\mathbf{x})}} = \frac{1}{1 + e^{-\mathbf{w}\cdot\mathbf{\Phi}(\mathbf{x})}} \tag{3.1}$$

C'est un modèle (log) linéaire : la décision de classification dépend d'une combinaison linéaire entre le vecteur de paramètres $\mathbf{w}$ et l'exemple à classer $\Phi(x, y)$. Dans le cas binaire, la classe est prédite selon la valeur de $h(x)$: la classe positive est prédite si $h(x) \geq 0,5$, donc quand le classifieur est à plus de 50% certain que la classe positive est la classe correcte, sinon la classe négative est prédite. La fonction logit a une valeur supérieure à $0,5$ quand son argument est supérieur à 0, on retrouve donc une décision fondée sur le signe du produit scalaire $\mathbf{w} \cdot \mathbf{\Phi}(\mathbf{x})$ comme pour d'autres classifieurs linéaires. L'apprentissage doit permettre de déterminer le meilleur vecteur de paramètres $\hat{\mathrm{w}}$, c'est celui qui maximise la log-vraisemblance sur les données d'entraînement, la vraisemblance étant donnée dans l'équation (3.2), cela correspond à maximiser la fonction de coût (3.3).

$$L(w) = \prod_{i=1}^{n} h(x_i)^{y_i}(1 - h(x_i))^{1-y_i} \tag{3.2}$$

$$\hat{\mathrm{w}} = \mathrm{argmax}_w \frac{1}{n} \sum_{i=1}^{n} y_i log(h(x_i)) + (1 - y_i) log(1 - h(x_i)) + \frac{1}{2\sigma^2} \sum_{j=1}^{d} w_j^2 \tag{3.3}$$

La vraisemblance est la probabilité que donne le modèle aux données d'entraînement. Plus elle est haute, plus le modèle représente fidèlement les données, avec un risque de sur-apprentissage. Pour gérer cette difficulté, on ajoute un terme de régularisation (le dernier terme dans l'équation (3.3)) qui va forcer les paramètres à rester petits pour éviter de construire une solution, une surface de séparation, trop complexe, donc trop attachée aux données. La régularisation dépend de l'hyper-paramètre λ, la force de la régularisation, et du type de norme utilisée : la formule (3.3) correspond à l'utilisation de la régularisation par une prior gaussienne, fondée sur la norme L_2 (norme euclidienne) au carré, le coefficient σ^2 correspondant à la variance de cette prior. Plus

on augmente le terme $\lambda = \frac{1}{2\sigma^2}$, plus on augmente la force de la régularisation. Il est également possible d'utiliser une régularisation se fondant sur la norme L_1 (somme des valeurs absolues des poids du vecteur) et un coefficient λ, le terme de régularisation a alors la forme : $\lambda \sum_{j=1}^{d} |w_j|$. Cette seconde forme a l'avantage de réduire le nombre de paramètres en favorisant des poids nuls. Le passage au multiclasse correspond à chercher la classe ayant la plus grande probabilité, la fonction de prédiction est donnée en 3.4. L'optimisation se fait de manière similaire au cas binaire.

$$h_y(x) = \text{argmax}_{y \in \mathcal{Y}} \frac{e^{\mathbf{w} \cdot \mathbf{\Phi}(\mathbf{x}, \mathbf{y})}}{\sum_{y' \in \mathcal{Y}} e^{\mathbf{w} \cdot \mathbf{\Phi}(\mathbf{x}, \mathbf{y}')}} \tag{3.4}$$

Différents algorithmes peuvent être utilisés pour estimer les paramètres comme la méthode d'optimisation dite à mémoire limitée (*limited memory BFGS*) implémentée dans MegaM [33] ou la méthode dite *Dual Coordinate Descent* (Yu et al., 2011) implémentée dans Scikit-Learn (Pedregosa et al., 2011).

Les études existantes ont testé plusieurs algorithmes en concluant généralement à la supériorité de l'algorithme bayésien naïf (Pitler et al., 2009 ; Rutherford et Xue, 2014), bien que, dans leur dernière étude, Rutherford et Xue (2015) utilisent par contre un algorithme par maximum d'entropie. C'est également le cas dans (Lin et al., 2009). Nous avons trouvé que, quand les paramètres de l'algorithme par maximum d'entropie sont correctement optimisés, les résultats sont en fait similaires à ceux obtenus avec l'algorithme bayésien naïf, si ce n'est supérieur. Notons de plus que ce dernier ne peut gérer des données contenant des valeurs négatives ce qui est le cas lorsque l'on utilise des plongements lexicaux dans le chapitre 5.

3.4.3 Résultats avec différents jeux de traits

Dans cette partie, nous donnons des résultats de référence en comparant différents jeux de traits pour les différents niveaux en les comparant, quand cela est possible, à des résultats présentés dans des études existantes. Rappelons que comme nous l'avons déjà dit, peu d'études sont comparables entre elles.

Nous avons choisi de fixer un jeu de traits minimal, correspondant à celui utilisé par Rutherford et Xue (2014) hors les paires de mots regroupés en clusters Brown, les règles de production et les traits de coréférence. Ce groupe de traits correspond à des traits repris d'études précédentes, notamment dans (Pitler et al., 2009 ; Park et Cardie, 2012). Cependant, dans ces précédentes études, chaque classifieur binaire correspond à une combinaison de traits différente, ce qui ne peut être mis en place en multiclasse. Nous choisissons de fusionner ces différents groupes de traits qui correspondent aux catégories suivantes, telles que nous les avons décrites en section 3.3.2 :

— **Nombre, pourcentage, dollars** : nombre de fois où l'on rencontre un nombre, une date (au sens d'une année sur quatre chiffres), un pourcentage ou un montant en dollars dans chaque argument, et somme de ces fréquences sur les deux arguments (traits continus),

— **Catégories sémantiques** : catégorie sémantique des verbes (la catégorie morpho-syntaxique contient "V") selon le lexique *General Inquirer* dans chaque argument et produit cartésien de ces catégories (traits nominaux binarisés),

— **Polarité** : nombre de mots de polarité positive, négative, positive niée, négative niée, neutre dans chaque argument et somme de ces fréquences sur les deux arguments. L'information de polarité est récupérée à partir du lexique *MPQA*, le fait qu'un mot nie une polarité est établi à partir du *General Inquirer* en regardant uniquement le mot précédent (traits continus),

33. http://www.umiacs.umd.edu/~hal/megam/version0_3/

— **Modalité** : trait binaire représentant la présence d'un modal (catégorie "MD") dans chaque argument et dans les deux. Nous avons trouvé que l'ajout des lemmes des modaux avait tendance à faire baisser les scores (traits binaires),

— **Verbes** : longueur des groupes verbaux et nombre de verbes appartenant à la même classe de Levin (traits continus),

— **Premier, dernier, trois premiers** : premier, dernier, et trois premiers mots de chaque argument, au sens des tokens, la casse est laissée intacte (traits nominaux binarisés).

Nous appelons ce jeu de traits **base**. Dans les résultats de référence ci-après, nous ajoutons à ces traits les règles de production syntaxiques (ensemble **base+synt**) telles qu'introduites par LIN et al. (2009) et/ou les traits de paires de mots (ensemble **base+lex**) tels qu'introduits par MARCU et ECHIHABI (2002). Nous évaluons la représentation de référence considérée comme la plus performante (traits issus de l'analyse syntaxique en dépendances) et nous vérifions si, avec notre configuration, la simplification de la représentation simplement obtenue par une lemmatisation ou une stemmatisation reste peu pertinente. Nous testons donc également les ensembles de traits restreints suivants :

— **Paires de mots** : produit cartésien en token sur les arguments (traits nominaux binarisés),

— **Paires de lemmes** : produit cartésien en lemmes sur les arguments, la lemmatisation est effectuée avec NLTK (traits nominaux binarisés),

— **Paires de stems** : produit cartésien en stems sur les arguments, la stemmatisation est effectuée avec NLTK (traits nominaux binarisés),

— **Règles syntaxiques** : règles de production, les règles sont obtenues à partir de l'analyse syntaxique manuelle fournie par la couche d'annotation du *Penn TreeBank* (traits nominaux binarisés).

Nous avons trouvé qu'ajouter des informations de type temps des verbes, chevauchement en mots ou longueur des arguments n'apportait pas d'améliorations, du moins telles qu'implémentées traditionnellement.

Toutes les performances rapportées dans cette section sont obtenues avec un classifieur par régression logistique implémenté dans le module Scikit-Learn (PEDREGOSA et al., 2011) [34]. Nous optimisons les hyper-paramètres de l'algorithme sur l'ensemble de développement en nous fondant sur le score de F_1 dans le cas de classifieurs binaires et de F_1 macro-moyenné en multiclasse. Ces hyper-paramètres correspondent à la norme de régularisation, entre L_1 et L_2, et à la force de cette régularisation (ici précisément $\frac{1}{\lambda}$, de plus petites valeurs pour le paramètre correspondent à une plus forte régularisation) comme présenté dans la section 3.4.2. $\lambda \in \{0,001,\ 0,005,\ 0,01,\ 0,1,\ 0,5,\ 1,\ 5,\ 10,\ 100\}$. Nous optimisons également un seuil en fréquence sur les traits $t \in \{1,\ 2,\ 5,\ 10,\ 15,\ 20\}$.

Rappelons la proportion d'exemples de chaque classe dans l'ensemble de test au niveau 1 : *Temporal* correspond à $6,5\%$ des données, *Contingency* à $26,4\%$ des données, *Comparison* à $14,0\%$ des données et *Expansion* à $53,1\%$ des données. Un système multiclasse de niveau 1 classifiant tous les exemples comme appartenant à la classe majoritaire *Expansion* correspond à un score de micro-exactitude de $53,1\%$ et à une macro-F_1 de $17,3\%$. Dans ce cas, comme dans le cas d'un classifieur binaire construit sur cette relation, la classe *Expansion* reçoit un score de F_1 de $69,4\%$. Pour le niveau 2, on a également un déséquilibre assez fort mais un classifieur majoritaire correspond à des performances bien plus basses avec 11 classes (exactitude de $25,9\%$ et la relation majoritaire *Cause* correspond à un score de F_1 de $51,6\%$). La proportion d'exemples de chaque classe est indiquée dans le tableau 3.5.

34. http://scikit-learn.org/stable/index.html

	Temporal		*Contingency*		*Comparison*		*Expansion*	
	P	F_1	P	F_1	P	F_1	P	F_1
Paires de mots (PM)	17,3	20,9	40,4	49,9	28,6	35,8	62,5	59,4
Paires de lemmes	16,8	19,6	37,9	48,4	16,9	26,2	62,5	59,4
Paires de stems	26,3	18,9	39,6	49,7	24,7	32,8	54,7	57,2
(PITLER et al., 2009) PM	-	16,2	-	43,8	-	21,0	-	63,8
(LI et NENKOVA, 2014b) PM	-	14,1	-	40,6	-	22,7	-	53,6
Règles syntaxiques (RS)	16,4	16,3	39,0	50,3	24,7	35,2	66,1	61,7
(PARK et CARDIE, 2012) RS	-	21,0	-	47,8	-	30,0	-	69,6
(LI et NENKOVA, 2014a) RS	16,9	20,6	35,4	47,6	19,9	28,4	43,9	57,4
(LI et NENKOVA, 2014b) RS	-	15,5	-	47,1	-	27,6	-	55,4
base	12,0	19,1	39,8	50,3	25,2	**36,6**	60,5	61,0
base+lex	28,1	25,6	44,0	**54,1**	28,6	35,7	65,4	**63,6**
base+synt	23,0	**25,8**	41,5	52,6	23,7	33,8	64,0	60,3
base+lex+synt (BLS)	37,1	25,2	39,6	49,5	27,1	**36,6**	67,1	63,3
(RUTHERFORD et XUE, 2014) ∼BLS	18,5	28,7	44,5	**54,4**	27,3	39,7	59,6	**70,2**
(RUTHERFORD et XUE, 2015) ∼BLS	-	24,6	-	53,9	-	38,0	-	67,9
(RUTHERFORD et XUE, 2015) best	-	**33,3**	-	53,8	-	**41,0**	-	69,4

Table 3.6.: Résultats de référence pour le niveau 1 en binaire pour différents jeux de traits : précision (« P ») et F_1 par relation. Nous rapportons les résultats des études précédentes correspondant à des jeux de traits similaires ainsi que les meilleurs résultats à ce jour dans cette configuration (RUTHERFORD et XUE, 2014) et (RUTHERFORD et XUE, 2015) best. Nous mettons en gras les meilleurs résultats que nous avons obtenus et les meilleurs scores de l'état de l'art. « - » indique que ce score n'est pas rapporté.

3.4.3.1 Classifieurs binaires au niveau 1

Nous rapportons dans la table 3.6 les scores obtenus pour des classifieurs binaires au niveau 1 (voir équation 3.1). Comme l'ont montré de précédentes études, la réduction de l'éparpillement des traits de type paires de mots passant par l'utilisation de stems ou de lemmes entraîne plutôt une dégradation, parfois assez importante, des performances.

On observe qu'en général nous obtenons des résultats supérieurs à ceux rapportés dans les états de l'art pour les sous-ensembles de traits paires de mots et règles syntaxiques, ceci étant dû à l'optimisation du filtre en fréquence, des hyper-paramètres de l'algorithme et probablement à l'utilisation de poids sur les instances plutôt qu'un sous-échantillonnage comme dans (PITLER et al., 2009). Pour les règles syntaxiques, on observe cependant une baisse pour *Temporal* et *Expansion* en termes de F_1. Pour *Temporal*, notons que les scores obtenus en utilisant le même filtre en fréquence que (LI et NENKOVA, 2014a), donc un filtre en fréquence de 5, et les paramètres par défaut de l'algorithme (norme L_2 et $\lambda = 1$) sont supérieurs : on obtient alors un score de F_1 de $23,4\%$ et une précision de $19,4\%$ [35]. L'optimisation de tous ces hyper-paramètres peut éventuellement dégrader les scores, c'est le cas si l'écart entre résultats sur l'ensemble de développement et sur l'ensemble d'évaluation sont assez éloignés. Nous considérons cependant qu'il est plus rigoureux d'optimiser le modèle plutôt que de fixer des valeurs *a priori*, en général l'optimisation permet une amélioration des performances. Pour *Expansion*, les résultats ne sont pas vraiment comparables car nous n'incluons pas les EntRel contrairement à PARK et CARDIE (2012).

Nous obtenons également des résultats assez proches de ceux présentés dans les études existantes utilisant un jeu de traits proche de **base+lex+synt** (systèmes « ∼ BLS »). Rappelons que dans ces études les paires de mots sont remplacées par des codes de clusters. Nous pensons que les

35. Rappelons que dans le cadre d'une recherche des meilleurs paramètres, on observe généralement une différence entre les performances sur l'ensemble de développement et d'évaluation : ainsi le meilleur modèle obtenu, en termes de performances sur l'ensemble de développement, n'est pas forcément celui qui conduit aux meilleures performances sur l'ensemble d'évaluation.

	Macro-prec	Macro-F_1	Exactitude
Paires de mots	42,1	38,5	53,7
Paires de lemmes	40,7	38,3	51,6
Paires de stems	37,1	36,1	49,9
Règles syntaxiques	41,4	**41,7**	51,0
base	39,1	39,2	49,9
base+lex	43,7	41,2	53,9
base+synt	39,4	39,9	49,4
base+lex+synt (BLS)	42,3	40,7	52,8
(RUTHERFORD et XUE, 2015) ∼BLS	-	38,4	55,0
(RUTHERFORD et XUE, 2015) best	-	40,5	**57,1**

Table 3.7.: Résultats de référence pour le niveau 1 en multiclasse pour différents jeux de traits, scores de de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »).

différences observées viennent cependant plutôt de l'utilisation de traits de coréférence, ce qui semble notamment profiter à *Expansion*. Les résultats sont généralement inférieurs aux scores état de l'art présentés dans (RUTHERFORD et XUE, 2015) (système « best ») qui mettent en jeu des données artificielles.

Le jeu de traits de type paires de mots permet d'obtenir des résultats assez hauts, seulement légèrement inférieurs à ceux obtenus avec le jeu de traits **base**. L'ajout à ce dernier d'informations syntaxiques et/ou lexicales permet d'obtenir les meilleurs résultats. En particulier, l'utilisation des paires de mots (**base+lex**) suffit voire permet des performances supérieures par rapport aux traits syntaxiques (**base+synt**) alors que ces derniers nécessitent l'utilisation d'un analyseur syntaxique. L'utilisation combinée de tous ces traits (**base+lex+synt**) améliore peu ou pas les scores obtenus en se limitant à l'ajout des paires de mots (**base+lex**).

3.4.3.2 Classifieur multiclasse au niveau 1

Nous rapportons dans la table 3.7 les résultats obtenus en multiclasse pour le niveau 1 de relation (voir équation 3.4). Seuls RUTHERFORD et XUE (2015) donnent des résultats par classe en multiclasse pour ce niveau. Rappelons que leur jeu de traits est similaire au jeu **base+lex+synt** sauf que les paires de mots sont remplacées par des paires de codes de clusters Brown et qu'ils utilisent des traits de coréférence. Notons également que ces auteurs optimisent le score d'exactitude et non la macro-F_1 donc les performances ne sont pas tout à fait comparables. RUTHERFORD et XUE (2015) (système « ∼BLS ») obtiennent un score d'exactitude supérieur à notre système correspondant à la représentation la plus similaire (**base+lex+synt**), nous obtenons un score de F_1 macro-moyenné supérieur. Cependant la relative proximité des scores semble indiquer que l'utilisation des clusters *Brown* n'entraîne pas une amélioration par rapport aux paires de mots bruts, nous reviendrons sur ce point dans le chapitre 5.

Comme on avait pu l'observer en binaire, la généralisation apportée par les lemmes ou les stems n'améliore pas les performances en termes de macro-F_1 avec une baisse particulièrement importante avec les stems. Les règles syntaxiques permettent d'obtenir ici des performances meilleures que celles obtenues avec les paires de mots, conclusion similaire à celle rapportée dans des études en multiclasse pour le niveau 2. Nous pensons cependant que le problème ne vient pas du type de représentation mais du problème d'éparpillement plus important pour les paires de mots.

Le tableau 3.8a reprend les scores par classe pour les traits de type paires de mots au niveau 1 en multiclasse. Comme on peut le constater , la relation *Temporal* est la moins bien identifiée, et, malgré la pondération des exemples, le système a tendance à prédire la classe *Expansion*.

Nous rapportons dans la table 3.8b les scores par classe pour le jeu de traits **base+lex+synt**. L'utilisation de ce groupe de traits plus diversifié permet une amélioration pour toutes les relations et notamment pour *Temporal*. Par comparaison, nous rapportons également les résultats présentés dans (RUTHERFORD et XUE, 2015), la dernière étude en date à ce niveau, en présentant leur modèle de référence (tableau 3.8c) et leur meilleur système (tableau 3.8d). Leur système de référence correspond globalement comme nous l'avons dit au jeu de traits **base+lex+synt**. Leur meilleur modèle est obtenu avec l'ajout de données de type explicite, système décrit en section 3.3.4.1. Notre modèle de référence avec les traits **base+lex+synt** correspond à des performances comparables voire supérieures à leur meilleur modèle ce qui peut être dû à l'optimisation d'hyper-paramètres différents et au fait que nous optimisons un score macro-moyenné, le meilleur système n'est donc pas celui qui classe le plus d'exemples correctement mais qui identifie bien globalement toutes les relations ce qui explique notamment la baisse de F_1 pour la relation majoritaire.

(a) Scores par classe (paires de mots).

Relation	P	R	F_1
Temp	22,6	17,6	19,8
Cont	47,3	38,8	42,6
Compa	38,6	18,5	25,0
Exp	59,7	74,8	66,4

(b) Scores par classe (**base+lex+synt**).

Relation	P	R	F_1
Temp	24,7	32,3	**28,0**
Cont	48,0	48,0	**48,0**
Compa	40,4	27,4	32,6
Exp	64,3	67,4	65,8

(c) Scores par classe : système de référence de (RUTHERFORD et XUE, 2015).

Relation	P	R	F_1
Temp	26,3	9,1	13,5
Cont	46,5	41,8	44,0
Comp	39,8	22,8	29,0
Expa	60,8	75,1	67,2

(d) Scores par classe : meilleur système de (RUTHERFORD et XUE, 2015).

Relation	P	R	F_1
Temp	38,5	9,1	14,7
Cont	49,3	39,6	43,9
Comp	44,9	27,6	**34,2**
Expa	61,4	78,8	**69,1**

Table 3.8.: Scores de précision (« P »), rappel (« R ») et F_1 par relation pour les systèmes de référence sur l'anglais en multiclasse au niveau 1 : traits paires de mots, ensemble **base+lex+synt** et résultats rapportés par RUTHERFORD et XUE (2015). Les relations de niveau 1 sont : *Temporal* (« *Temp* »), *Contingency* (« *Cont* »), *Comparison* (« *Comp* ») et *Expansion* (« *Expa* »).

3.4.3.3 Classifieur multiclasse au niveau 2

Nous rapportons dans la table 3.9 les résultats pour différents jeux de traits au niveau 2 en multiclasse. Nous incluons les résultats obtenus par LIN et al. (2009) pour les sous-ensembles de traits utilisés dans cette étude ainsi que les résultats finaux. Ces auteurs optimisent également l'exactitude. On note que nos résultats pour le jeu de traits syntaxiques sont légèrement inférieurs en termes d'exactitude micro-moyennée à ceux obtenus par LIN et al. (2009), ces auteurs ne donnent pas les scores par classe pour ce jeu de traits donc nous ne pouvons pas calculer la macro-F_1. Avec un filtre 5 nous obtenons $36,7\%$ d'exactitude donc similaire à celle rapportée.

Pour le jeu de trait de type paires de mots, nous obtenons par contre une exactitude bien supérieure, ce qui peut s'expliquer par l'optimisation du filtre en fréquence et l'utilisation d'un schéma de pondération d'instance. Encore une fois, ce jeu de traits permet d'obtenir de bonnes performances, similaires à celles obtenues par LIN et al. (2009) bien que le jeu de traits soit bien moins riche ce qui montre l'importance de ces paramètres. Comme dans le cas binaire au niveau 1, nous obtenons de meilleures performances en ajoutant les traits de type paires de mots à l'ensemble de traits de référence (**base+lex**) qu'en ajoutant les règles syntaxiques (**base+synt**). L'ajout de ces deux types d'information permet d'obtenir les meilleurs scores avec une exactitude micro-moyennée de $42,8\%$ assez proche des $43,6\%$ rapportés dans (JI et EISENSTEIN, 2014a), les meilleurs scores obtenus à ce jour, avec cependant une représentation bien plus simple. Rappelons que ces auteurs ne rapportent

	Macro-prec	Macro-F_1	Exactitude
Paires de mots (PM)	35,5	23,1	39,9
Paires de lemmes	37,4	23,3	38,9
Paires de stems	**37,5**	23,2	38,9
(Lin et al., 2009) PM (filtre 5)	-	-	30,3
(Lin et al., 2009) PM (filtre IM)	-	-	32,9
Règles syntaxiques (RS)	22,3	22,3	36,2
(Lin et al., 2009) RS (filtre 5)	-	-	36,7
(Lin et al., 2009) RS (filtre IM)	-	-	38,4
base	21,4	21,6	34,3
base+lex	33,4	26,0	41,2
base+synt	23,5	23,1	36,5
base+lex+synt	31,0	**26,7**	**42,8**
(Lin et al., 2009)	34,1	20,4	40,2
(Ji et Eisenstein, 2014a)	-	-	**43,6**

Table 3.9.: Résultats de référence pour le niveau 2 en multiclasse pour différents jeux de traits, scores de de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »).

pas les scores par classe et qu'ils n'utilisent ni filtre en fréquence ni aucune méthode destinée à gérer le déséquilibre des classes.

Dans le tableau 3.10a, nous rapportons les scores par classe pour le jeu de traits paires de mots. Comme on peut le voir, trois relations reçoivent un score nul, ce sont les trois relations les moins représentées dans les données. Deux autres relations sous-représentées, *Synchronous* et *Concession*, correspondent à un score bas. Dans le tableau 3.10b, nous rapportons les scores par classe pour le jeu de traits **base+lex+synt**. On observe que les cinq relations sous-représentées reçoivent toujours un score nul ou très bas. Nous donnons par comparaison les scores rapportés dans (Lin et al., 2009) dans le tableau 3.10c. Rappelons que la comparaison n'est pas directe car ces auteurs conservent les annotations multiples et n'utilisent pas les mêmes sections d'évaluation. Bien que nous n'utilisions pas de traits provenant d'une analyse en dépendance ni de traits de contexte, nos résultats sont comparables voire supérieurs. Quatre des cinq relations sous-représentées correspondent ici à un score nul. Nous pensons que les améliorations que nous obtenons sont notamment dues à l'utilisation d'une stratégie destinée à gérer le déséquilibre des classes encore plus fort à ce niveau de relation. Nos scores sont généralement supérieurs à l'exception de la relation *List*. Lin et al. (2009) indiquent disposer de 30 exemples pour cette relation mais ce nombre prend en compte les annotations multiples, il est possible que la stratégie d'évaluation prenant en compte ces annotations multiples favorise cette relation. Avec un rappel de 13%, cette relation est cependant assez mal identifiée également dans cette étude. Il est cependant possible qu'elle profite dans (Lin et al., 2009) notamment des traits contextuels. C'est une relation assez spécifique qui correspond à la mise en lien de plusieurs unités déjà liées généralement par une relation de type *Conjunction*. La prise en compte de la structure générale doit donc améliorer son identification.

(a) Scores par classe (paires de mots).

Relation	P	F_1
Asyn	$29,4$	$22,7$
Syn	$100,0$	$\mathbf{13,3}$
Caus	$40,1$	$47,6$
Prag Caus	$0,0$	$0,0$
Contr	$33,6$	$32,4$
Conc	$50,0$	$\mathbf{10,5}$
Conj	$41,6$	$41,3$
Inst	$57,9$	$51,6$
Rest	$37,5$	$34,9$
Alt	$0,0$	$0,0$
List	$0,0$	$0,0$

(b) Scores par classe (**base+lex+synt**).

Relation	P	F_1
Asyn	$28,9$	$\mathbf{26,3}$
Syn	$50,0$	$12,5$
Caus	$44,9$	$50,2$
Prag Caus	$0,0$	$0,0$
Contr	$33,8$	$\mathbf{34,1}$
Conc	$16,7$	$8,7$
Conj	$43,3$	$\mathbf{44,4}$
Inst	$60,5$	$\mathbf{58,1}$
Rest	$40,9$	$\mathbf{37,7}$
Alt	$22,2$	$\mathbf{22,2}$
List	$0,0$	$0,0$

(c) Scores par classe (LIN et al., 2009).

Relation	P	F_1
Asyn	50	13
Syn	0	0
Caus	39	**51**
Prag Caus	0	0
Contr	61	15
Conc	0	0
Conj	30	38
Inst	67	49
Rest	48	35
Alt	0	0
List	80	**23**

Table 3.10.: Scores de précision (« P ») et F_1 par relation pour les systèmes de référence sur l'anglais en multiclasse au niveau 2 : traits paires de mots, ensemble **base+lex+synt** et résultats rapportés par LIN et al. (2009), optimisation de la F_1 macro-moyennée. Les relations de niveau 2 sont : *Asynchronous* (« *Asyn* »), *Synchronous* (« *Syn* »), *Cause* (« *Caus* »), *Pragmatic Cause* (« *Prag Caus* »), *Contrast* (« *Contr* »), *Concession* (« *Conc* »), *Conjunction* (« *Conj* »), *Instantiation* (« *Inst* »), *Restatement* (« *Rest* »), *Alternative* (« *Alt* ») et *List* (« *List* »).

4 Adaptation des données explicites aux données implicites

Sommaire

Nous avons présenté dans le chapitre précédent les systèmes construits pour l'analyse discursive complète des documents ainsi que les systèmes dédiés aux sous-tâches de segmentation et d'identification des relations. Les systèmes complets obtiennent encore des performances modestes. Nous avons vu que, dans ces systèmes, l'identification automatique de la relation liant deux segments donne des scores relativement bas, c'est même généralement la sous-tâche correspondant aux plus mauvaises performances. HERNAULT et al. (2010) rapportent ainsi $47,7\%$ de F_1 sur le RST DT et MULLER et al. (2012a) $43,3\%$ sur ANNODIS en prenant en compte un jeu de relations similaire. Dans le système de LIN et al. (2014) sont séparées relations explicites, correspondant à $86,77\%$ de F_1, et non explicites pour lesquelles le score obtenu est de l'ordre de 40%. Améliorer l'identification des relations est donc une étape nécessaire pour produire des systèmes complets performants. En particulier, le système de LIN et al. (2014) montre que la difficulté vient essentiellement des relations implicites. Les autres corpus ne distinguant pas exemples explicites et implicites, on peut supposer que les performances générales assez basses sont également dues à ce type d'exemple.

Cette observation a conduit à différentes études centrées sur l'identification des relations discursives implicites en particulier sur le PDTB, études se divisant en deux grands courants. Le premier ensemble d'études repose sur l'amélioration de la représentation des données à l'aide de ressources

construites à la main, PARK et CARDIE (2012) et LIN et al. (2009) rapportant les meilleurs scores dans ce cadre respectivement pour le niveau 1 et 2 de relations dans le PDTB. Cette stratégie a ses limites. Améliorer les performances dans ce cadre nécessite d'ajouter encore de nouvelles ressources, ressources coûteuses à construire. De plus, les informations utilisées ne sont pas disponibles pour la plupart des langues. Ces difficultés ont conduit au développement d'un second courant d'études fondées sur une forme de non supervision, l'idée étant de construire des systèmes utilisant des informations supplémentaires acquises cette fois automatiquement. Les premières études dans ce cadre ont cherché à augmenter la taille des données disponibles en utilisant des exemples annotés automatiquement à partir des connecteurs, ou exemples artificiels. C'est cette stratégie que nous proposons d'explorer dans ce chapitre. Les expériences sur le français décrites dans ce chapitre ont été publiées dans (BRAUD et DENIS, 2014b ; BRAUD et DENIS, 2014a).

L'utilisation de données explicites pour l'identification des relations implicites est une idée relativement ancienne (MARCU et ECHIHABI, 2002). Elle pose cependant un certain nombre de questions quant à sa validité. Notamment, elle se fonde sur deux grandes hypothèses : l'hypothèse de redondance du connecteur avec son contexte, c'est-à-dire le fait que l'on peut encore inférer la relation une fois le connecteur supprimé, et l'hypothèse de similarité entre exemple explicite (ou plutôt artificiel) et exemple implicite, qui correspond à la possibilité de généralisation d'un modèle entraîné sur des données du premier type mais évalué sur des données du second type. Nous décrivons cette stratégie dans la section 4.1 et détaillons les hypothèses sur lesquelles elle repose en commençant par l'hypothèse de redondance du connecteur en section 4.2. Nous nous intéressons ensuite à la question de la similarité entre les données que nous rapprochons des hypothèses de base d'un système d'apprentissage en section 4.3. Les sections suivantes sont consacrées aux expériences que nous avons menées. En section 4.4 nous décrivons les corpus artificiels que nous avons construits pour le français et l'anglais. Nous présentons ensuite en section 4.5 les stratégies mises en place ainsi que les résultats obtenus sur ANNODIS et sur le PDTB.

4.1 Méthode

Dans cette section, nous introduisons la méthode d'identification des relations implicites qui s'appuie sur les données explicites en revenant sur son principe (section 4.1.1) et sur les hypothèses qu'elle fait (section 4.1.2). Nous détaillons ces hypothèses dans les sections suivantes.

4.1.1 Principe

Comme nous l'avons vu dans le chapitre précédent, cette stratégie a d'abord été proposée par MARCU et ECHIHABI (2002). Le modèle utilisé par ces auteurs reposait uniquement sur les paires de mots dans le produit cartésien sur les arguments. L'exemple (37) serait ainsi représenté par l'ensemble des paires de mots telles qu'une paire correspond à un mot du premier argument et à un mot du second argument. L'intuition sur laquelle repose cette représentation est que certaines paires peuvent signaler une relation comme la paire « *partielle,totale* » qui est un bon indicateur d'une relation de type contrastive dans (37) issu du corpus ANNODIS.

(37) [Pour la majeure partie du pays, il s' agira d' une éclipse **partielle**.] [Seule une bande du territoire d' environ 110 km de large comprenant la Lorraine, bénéficiera de l' éclipse **totale**.]

Comme cette représentation pose le problème d'éparpillement décrit dans le chapitre précédent, MARCU et ECHIHABI (2002) proposent d'enrichir leur système de nouvelles données acquises automatiquement. Afin d'obtenir des données brutes annotées en relations, ils choisissent d'utiliser

les connecteurs comme étiquette d'une relation. En effet, que les exemples soient marqués par un connecteur ou non, le lien discursif qui les unit est du même type, une relation sémantico-pragmatique liant contenu sémantique des arguments et/ou acte de parole exprimé dans les arguments. On a donc le même type d'étiquette ou de classe, c'est-à-dire le même ensemble de sortie pour un système statistique. De plus ces relations mettent en lien des segments textuels similaires. Même si l'on pourra noter des différences entre ces segments, comme l'utilisation de modes différents pour les verbes mis en jeu, ce sont des unités élémentaires de discours essentiellement de type clausal. L'ensemble d'entrée est donc similaire.

La principale différence entre exemples explicites et implicites est bien sûr la présence ou l'absence d'un connecteur discursif donc d'un mot spécifique capable de déclencher une relation ou de contraindre les relations possibles. On l'a vu, c'est un indice fort puisqu'il permet à lui seul d'identifier la relation dans la majorité des cas. L'idée est donc de supprimer cet indice fort, peu ou pas ambigu et spécifique des exemples explicites. On obtient deux segments discursifs liés par une relation connue, celle que déclenchait le connecteur, mais qui n'est plus marquée par ce connecteur. Ce sont donc des exemples similaires aux exemples implicites mais créés artificiellement contrairement aux exemples originellement implicites, qui ne contenaient pas de connecteur.

Rappelons ici la terminologie que nous utilisons pour désigner les différents ensembles de données. On appelle *exemple implicite artificiel* un exemple implicite qui était à l'origine explicite et *exemple implicite naturel* un exemple implicite qui n'a subi aucune modification, qui est un exemple implicite attesté. Les exemples (implicites) naturels correspondent aux exemples implicites manuellement annotés dans les corpus, on pourra parler parfois d'exemples (naturels) manuels. Les exemples (implicites) artificiels proviennent d'exemples explicites qui sont soit annotés également manuellement (exemples artificiels manuels), soit extraits automatiquement à partir de données brutes et d'heuristiques ou de modèles (exemples artificiels automatiques). Dans le cas d'exemples artificiels manuels, la relation est annotée. Pour les exemples artificiels automatiques, il faut l'identifier. En général, dans les études existantes, elle correspond à la relation la plus fréquente pour le connecteur considéré.

Ainsi, dans l'exemple (38a), issu du corpus *Est Républicain*, le connecteur *cela dit* marque une relation *Contrast* entre les deux phrases. On construit un exemple artificiel (38b) [1] en supprimant ce connecteur et en annotant l'exemple de type implicite ainsi créé avec la relation *Contrast*. On note dans cet exemple les paires de mots « *comique, drame* » et « *drôle, drame* » qui signalent une opposition. L'utilisation de ces données artificielles suppose donc que l'on pourra trouver ces paires également dans des exemples implicites naturels. Cette méthode repose en fait sur plusieurs hypothèses qui reflètent les différences entre exemples explicites et implicites.

(38) a. [Elle était très comique, très drôle.] [*Cela dit*, le drame n'était jamais loin.] Exemple explicite attesté, Est Républicain

b. [Elle était très comique, très drôle.] (*contrast*) [, le drame n'était jamais loin.] Exemple explicite artificiel, Corpus artificiel

1. Notons ici le traitement de la ponctuation : nous n'avons pas défini d'heuristiques pour définir si un signe de ponctuation devait être supprimé en même temps que le connecteur. C'est pourquoi l'exemple artificiel contient ici une ponctuation non naturelle, avec une virgule à la suite d'un point.

4.1.2 Hypothèses faites par la méthode

L'utilisation de données explicites comme données supplémentaires dans le cadre d'un système d'identification des exemples implicites pose différents problèmes qui se reflètent dans les performances obtenues par les premiers systèmes utilisant ces nouvelles données. Notamment, PITLER et al. (2009) montrent que l'on obtient de meilleures performances en se limitant aux paires de mots présentes dans les données implicites. SPORLEDER et LASCARIDES (2008) étendent la représentation à des traits linguistiquement motivés et obtiennent des performances bien plus basses en utilisant les données artificielles qu'en se limitant aux données naturelles pourtant disponibles en bien moins grande quantité. Ces expériences montrent que les données sont de nature différente : si elles étaient similaires, l'apport de nouvelles données ne pourrait qu'améliorer la capacité de généralisation du modèle et donc les scores. Plus précisément, cela montre que l'on ne peut pas se servir des données artificielles directement comme données d'entraînement, il faut opérer une forme d'adaptation.

L'utilisation des données artificielles repose sur deux hypothèses fortes. Elle suppose que le connecteur est redondant avec son contexte et que les exemples créés artificiellement sont similaires aux exemples attestés :

1. Hypothèse de *redondance du connecteur* : la relation est toujours inférable une fois le connecteur supprimé, les effets de cette suppression sont donc négligeables et il doit exister des indices autres que le connecteur dans les arguments qui permettent d'identifier la relation.
2. Hypothèse de *similarité des données* : les données artificielles et naturelles sont suffisamment similaires pour permettre à un modèle construit sur les artificielles d'acquérir des informations permettant d'identifier la relation dans les naturelles.

Les performances des systèmes précédemment citées ainsi que nos propres expériences montrent que ces hypothèses sont trop fortes. Dans la section suivante, nous détaillons les problèmes posés par l'hypothèse de redondance en montrant qu'elle n'est pas vérifiée dans tous les cas mais que cela ne constitue pas, selon nous, un obstacle majeur à l'utilisation des données artificielles. Nous reviendrons ensuite sur l'hypothèse plus générale de similarité entre les données en montrant qu'elle n'est pas vérifiée et que l'utilisation des données artificielles nous inscrit dans le cadre de l'apprentissage avec données non identiquement distribuées.

4.2 Redondance du connecteur

On s'intéresse dans cette section à l'hypothèse de redondance du connecteur. On se pose en particulier la question suivante : est-ce que la suppression d'un connecteur a un effet particulier sur l'inférence de la relation qu'il marquait ? On se demande donc si un exemple artificiel est un « bon » exemple implicite au sens où il est acceptable et cohérent, et au sens où il met bien en jeu la relation précédemment marquée par le connecteur. La question qui concerne le fait qu'il reste ou non suffisamment d'information pour inférer la relation est bien sûr liée : si l'exemple artificiel répond aux caractéristiques précédentes, on peut inférer la relation donc il reste suffisamment d'information. Elle n'est pas tout à fait équivalente cependant puisque le cas où l'exemple artificiel n'est pas acceptable pour des raisons syntaxiques n'implique pas que la relation n'est plus inférable. En effet, dans ce cas, il peut rester suffisamment d'information pour un système automatique même si un humain rejetterait un exemple agrammatical. Rappelons que des études comme celles de (MARCU et ECHIHABI, 2002 ; SPORLEDER et LASCARIDES, 2008 ; PITLER et al., 2009) ont montré que l'hypothèse de redondance était vérifiée dans une partie des données puisque les systèmes

qu'ils présentent, entraînés et testés sur des données artificielles, obtiennent des performances largement supérieures à la chance.

L'hypothèse de redondance du connecteur avec son contexte est une hypothèse forte au sens où l'on peut penser que si un connecteur lexicalise la relation, alors il est suffisant et nécessaire. De manière à mieux évaluer l'hypothèse de redondance, nous envisageons les différents effets discursifs de la suppression d'un connecteur. Nous discuterons aussi brièvement des effets syntaxiques que nous incluons dans la seconde catégorie. Schématiquement, on identifie quatre cas possibles que nous détaillons dans les sections suivantes :

1. La suppression du connecteur a peu ou *pas d'effet* : la relation reste inférable en son absence, le connecteur est redondant avec son contexte (section 4.2.1).
2. La suppression du connecteur rend le *discours incohérent* : le discours devient inacceptable et ce non pas simplement à cause d'effets syntaxiques (section 4.2.2).
3. La suppression du connecteur entraîne une *modification* de la relation inférée : la relation lexicalisée par le connecteur n'est plus inférable mais une autre relation est inférée, le discours reste donc cohérent (section 4.2.4).
4. La suppression du connecteur entraîne une *difficulté* à inférer la relation : la relation reste inférable mais son inférence demande un effort important (section 4.2.4).

Dans cette section, nous nous intéressons aux effets de la suppression d'un connecteur selon les différentes catégories présentées précédemment. Pour cela, nous examinons des exemples tirés de nos données ou proposés dans diverses études. Nous verrons que l'hypothèse de redondance n'est pas vérifiée dans tous les cas ce qui implique que les données artificielles contiendront de « mauvais » exemples implicites, des exemples incohérents ou agrammaticaux ou, ce qui est plus grave dans un cadre d'apprentissage statistique, des exemples annotés avec une étiquette ne correspondant pas à la relation inférable par un humain. On aura donc une forme de bruit dans nos données et nous proposerons dans les sections suivantes des méthodes pour le gérer. Notons que nous prenons ici les connecteurs au sens où ils sont définis dans les ressources que nous utilisons. Nous ne remettons pas en cause la nature de connecteur des formes sélectionnées dans LexConn ou dans le PDTB, même si certains choix peuvent faire débat, par exemple *et* n'est pas toujours considéré comme un marqueur discursif du moins pas de la principale relation inférable comme c'est par exemple le cas pour le *et* intervenant dans une opposition (Corminboeuf, 2013).

Les études sur les connecteurs distinguent généralement deux notions : l'optionnalité d'un connecteur et sa redondance (Schourup, 1999). Ainsi, ce que Rutherford et Xue (2015) caractérisent comme une hypothèse répandue, c'est-à-dire le fait que la relation discursive demeure la même lorsque le connecteur est supprimé, n'est pas si populaire que ça. Les connecteurs peuvent être vus comme optionnels dans le sens où « ils n'élargissent pas les possibilités de relations sémantiques entre les segments qu'ils relient » (Schourup, 1999). Si le connecteur est supprimé, la relation qu'il déclenchait est toujours disponible. Mais les connecteurs ne sont jamais considérés globalement comme redondants car ils permettent de guider l'interprétation, de faciliter la compréhension (Liesbeth et al., 1999) et d'écarter des interprétations non désirées, ils contraignent l'interprétation. Cependant, les exemples montrent que dans certains cas le connecteur est bien complètement redondant et que dans d'autres il n'est absolument pas supprimable, sous peine de perdre l'inférence de la relation ou la cohérence. Ce contraste ne semble cependant pas être une caractéristique des connecteurs mais plutôt des relations qu'ils déclenchent liée à des principes cognitifs. Nous verrons également qu'entre ces deux extrêmes l'alternance explicite-implicite semble possible, tout en pouvant être parfois contrainte par le contenu des arguments.

Enfin, un dernier point à noter concerne les cas d'exemples implicites pour lesquels l'explicitation est impossible. Nous avons dit dans la section 2.4.2 que certaines relations étaient toujours implicites,

parce qu'aucun connecteur de ces relations n'avait été identifié. C'est par exemple le cas de la relation *Elaboration* incluse dans le jeu de relations utilisé dans ANNODIS. Certaines relations sont quant à elles très préférentiellement implicites, on ne trouve que peu d'exemples explicites dans les données. C'est par exemple le cas de la relation *Pragmatic Cause* dans le PDTB qui correspond à 69 exemples implicites contre seulement 8 explicites (voir tableau 2.2). Pour ces relations, nos modèles ne pourront donc véritablement tirer profit des exemples artificiels, ils ne pourront se fonder que sur les exemples implicites disponibles. Un autre cas intéressant concerne les paires de segments mettant en jeu une attribution dans le second argument, un problème discuté dans (HUNTER et DANLOS, 2014). Ainsi, dans l'exemple (39), il est impossible d'insérer le connecteur *parce que* entre les deux phrases. Il sera donc impossible de trouver des exemples de relations explicites similaires à cet exemple de relation implicite. Ceci correspond à une différence en termes distributionnels sur le type de paires de segments que l'on peut rencontrer dans les deux ensembles de données.

(39) Paul ne viendra pas à ma fête. Marie a dit qu'il avait quitté la ville.

4.2.1 Connecteur redondant

Dans certains cas, le connecteur est redondant c'est-à-dire supprimable sans effet sur la cohérence du discours ou l'inférence de la relation. C'est par exemple le cas dans les exemples (40a) et (40b). Cette possibilité semble dépendre essentiellement du contenu des segments liés et de préférences sur les relations inférables en l'absence de marques explicites. Deux principes ont été proposés pour expliquer les préférences des locuteurs. Le premier principe (SANDERS, 2005 ; ASHER et LASCARIDES, 1998) stipule qu'en l'absence de marquage explicite d'une relation, on a tendance à préférer construire la représentation la plus informative possible, à inférer le lien le plus riche c'est-à-dire un lien de causalité entre des évènements. Bien sûr, il ne s'agit que d'une préférence, l'inférence d'un lien de causalité dépend quand même du contenu des propositions et, au moins partiellement, de connaissances sur le monde. Il explique la redondance du connecteur *parce que* dans l'exemple (40b) mais si l'on modifie le contenu des propositions et notamment le temps des verbes, l'inférence d'un lien causal est bloquée comme dans l'exemple (40c). Cette préférence pour un lien causal est doublée d'un second principe correspondant à l'inférence d'une continuité temporelle par défaut : si aucun lien plus informatif n'est inférable, alors on infère un simple lien de narration entre les évènements dans l'ordre du texte ce qui est le cas dans les exemples (40a) et (40c). Dans l'exemple (40a), on a de plus un effet dû à nos connaissances sur des enchaînements d'actions habituelles. Dans l'exemple (40c), l'enchaînement de verbes au passé simple empêche d'identifier un ordre temporel non linéaire.

(40) a. L'avion atterrit (*et*) les passagers descendirent.

b. Paul tomba (*parce que*) Marie l'avait poussé.

c. Paul tomba. Marie le poussa.

Ces hypothèses sont appuyées par des expériences psycho-linguistiques. Ainsi, ASHER et LASCARIDES (1998) reprennent une étude portant sur le jugement de locuteurs à propos du lieu où les locations sont les moins chers à partir de phrases comme celles en (41). Même si les locuteurs savaient, d'après leurs connaissances du monde, que les loyers étaient moins chers à Belleville, la majorité d'entre eux a jugé que dans ce texte, le loyer moins cher dont il est question réfère à Passy. Ceci s'explique par le fait que les locuteurs ont préféré inférer une relation d'explication entre les deux phrases plutôt que de considérer le discours comme incohérent ou d'inférer un lien concessif qui nécessiterait l'insertion d'un connecteur comme *pourtant*.

(41) John a déménagé de Belleville à Passy.
Le loyer était moins cher.

Dans leur étude, SORIA et FERRARI (1998) demandent à des sujets d'identifier une relation, parmi *additive* (une forme de relation narrative), *Consequential* (une forme de relation causale) et *Contrastive*, dans des exemples où le connecteur est ou non supprimé (l'exemple restant grammatical). L'étude montre que l'identification est effectivement plus difficile sans connecteur. On passe de 72.6-88.9% à 42.7-64.3% d'identification correcte par relation. Les scores d'identification de la relation sans connecteur restent cependant au-delà de la chance comme pour les systèmes automatiques. Cependant, le statut des relations diffère : la relation *additive* semble prédite par défaut, car la plus souvent prédite par erreur, et la relation *Consequential* reste bien prédite ce qui reflète une préférence des locuteurs pour les liens causaux. On retrouve ces conclusions dans une étude menée sur le PDTB par ASR et DEMBERG (2013) : les exemples mettant en jeu un enchaînement temporel non linéaire des évènements sont plus souvent marqués que ceux pour lesquels l'ordre est linéaire, et les relations causales sont plus facilement implicites. Cela ne signifie pas que seules les relations temporelles et causales restent identifiables en l'absence de connecteur. Ainsi, dans l'étude de SORIA et FERRARI (1998), la relation de type contrastive correspond à une identification correcte de 42, 7% ce qui reste assez élevé.

Dans une étude de 28 exemples explicites de la relation *Explanation* sur ANNODIS [2], nous avons trouvé 20 cas où le connecteur nous a semblé tout à fait redondant, comme dans les exemples (42a), (42b), (42c) et (42d) [3]. Nous nous sommes particulièrement intéressée à cette relation parce qu'elle semblait la plus susceptible de redondance, de par les préférences des locuteurs pour l'inférence de ce lien, sans être pour autant une relation par défaut. Comme nous le verrons, la suppression du connecteur dans les exemples de cette relation peut cependant avoir des effets sur le discours. Notons que dans les exemples ci-dessous, d'autres indices semblent apparaître comme l'adverbe « *finalement* » en (42b). Concernant les deux autres exemples, c'est plutôt le contenu propositionnel qui nous incite à inférer un lien causal, on cherche une explication à l'assertion présente dans le premier segment : pourquoi la précision est-elle limitée ? pourquoi ce réseau était-il différent ? Ici, on pourrait se demander si la paire « *limitée, dépendante* » n'est pas un indice signalant le lien causal.

(42) a. [la précision de cette science est limitée] (*car*) [elle est dépendante des éléments osseux et matériels mis au jour au fur et à mesure des fouilles]
b. [Grosse journée pour les nombreux bénévoles de la section] (*puisque*)
[, finalement, 210 petits combattants ont répondu présents !]
c. [BITNET était différent d'Internet] (*parce que*) [c'était un réseau stocké puis transmis]
d. [Cette loi du silence règne] (*car*) [elle joue sur la peur que les non mafieux ont de la mafia (...)]

4.2.2 Cohérence du discours

Dans certains cas par contre, le connecteur est nécessaire : sa suppression peut rendre le discours incohérent. Cet effet a été noté par ASHER et LASCARIDES (2003) dans le cas des contrastes de type violation d'attente. Ainsi dans l'exemple donné par ces auteurs et repris en (43a) (nous traduisons)

2. Rappelons que les connecteurs ne sont pas annotés dans ce corpus. Nous avons donc projeté LexConn sur les données puis vérifié manuellement chaque exemple. Ainsi, le chiffre que nous donnerons plus loin sur le nombre de relations explicites de cette relation n'est pas 28, la correction manuelle ayant révélé qu'il y avait moins d'explicites.

3. Les trois premiers exemples correspondent à une relation *Evidence* dans le cadre de la RST, car le premier argument contient une déclaration plutôt que la description d'une situation. Nous n'avons pas pu conclure de manière définitive que le connecteur était plus facile à supprimer dans le cas de cette relation spécifique.

la suppression du connecteur *mais* conduit à un discours bizarre jugé incohérent. Par contre, dans les contrastes formels, où il n'y a pas de relation logique entre les conditions de vérité des arguments mais plutôt un contraste dû à une différence de contenu, le marqueur n'est pas nécessaire : dans l'exemple (43b) que nous traduisons également, la suppression du connecteur aboutit à une perte d'information, au sens où la mise en contraste des deux informations est perdue, mais le discours reste acceptable.

(43) a. John aime le sport. (*Mais*) il déteste le football.

b. John a les yeux verts. (*Mais*) Mary a les yeux bleus.

SPORLEDER et LASCARIDES (2008) trouvent environ 30% d'exemples de *Contrast* rendus incohérents par la suppression du connecteur. Les auteurs notent cependant que cet effet peut disparaître si le connecteur de contraste, ici *but*, est renforcé par un marqueur pouvant signaler *Contrast* comme *then again* dans l'exemple (44a). Bien sûr, si deux connecteurs sont utilisés, par exemple *mais pourtant* dans un exemple comme (43a), la suppression de l'un des deux laisse l'exemple cohérent. Nous verrons que l'heuristique que nous avons définie pour construire les données artificielles prend en compte ce cas : lorsque deux connecteurs sont identifiés, nous ne conservons pas l'exemple si ces deux connecteurs ont les mêmes arguments (ce qui serait le cas ici). Cette règle nous permet de nous assurer en partie que les exemples implicites artificiels ne contiennent vraiment pas de connecteur. Les cas non pris en compte sont ceux où le connecteur est suivi par une forme pouvant éventuellement marquer la relation mais qui n'appartient pas au lexique de connecteur utilisé, soit parce qu'il a été oublié, soit parce qu'il ne correspond pas aux critères définissant un connecteur.

(44) a. Manually adding best-fit curves to data plots can be laborious and prone to error. ((*But*) *then again*), Don Bradbury reviews TableCurve and still finds it has all the right lines.

Notons que cet effet, c'est-à-dire le fait que la suppression du connecteur conduit à un discours incohérent, est présent dans l'exemple que nous avons donné au début de cette section, provenant de nos données artificielles françaises, et repris en (45a) : on a bien affaire ici à un contraste de type violation d'attente. C'est également le cas dans l'exemple (45b) issu du corpus ANNODIS dans lequel ces deux types de contraste ne sont pas séparés. Nous avons donc dans nos données naturelles et artificielles des exemples pour lesquels le connecteur n'est pas supprimable, au sens de la cohérence du discours. Cependant, l'exemple naturel n'est pas inclus dans notre corpus d'évaluation puisque c'est un exemple explicite. La question est donc de savoir si le type d'information présent dans des exemples comme (43a), (45a) et (45b) est présent également dans des exemples implicites naturels, est-ce que, par exemple, on peut trouver la paire « *drôle, drame* » ou la paire « *aime, déteste* » dans un exemple de type contraste formel. Nous pensons que c'est possible au sens où le discours en (45c) semble acceptable dans un contexte spécifique similaire à celui qui rend acceptable l'exemple explicite (43b). Notons que, même si ce n'est pas le cas, ce genre de paires opposées a peu de chance d'indiquer une relation autre que *Contrast* donc il nous semble que l'inclusion de ces exemples ne devrait pas dégrader les performances du système. Il est cependant possible que le système établisse une corrélation entre un discours incohérent et la relation *Contrast*, alors que, bien sûr, il n'y a pas d'incohérence dans les exemples naturels.

(45) a. Elle était très comique, très drôle. (*Cela dit*), le drame n'était jamais loin.

b. La hulotte est un rapace nocturne *mais* elle peut vivre le jour.

c. ?Jean aime le sport. Mais Marie déteste le football.

Nous ajoutons dans cette catégorie les cas d'agrammaticalité provoqués par la suppression du connecteur. S'ils n'ont bien sûr pas de lien avec la question de la cohérence, ils correspondent également à des discours qui seraient rejetés comme bizarres ou mal formés par des locuteurs. Ces cas peuvent correspondre à des problèmes liés au mode des verbes comme dans l'exemple (46a) de type contrastif. Ils peuvent aussi s'appliquer dans le cas de l'usage cataphorique d'un pronom comme dans l'exemple construit (46b) que l'on peut opposer à l'exemple (46c) où le connecteur *parce que* est au contraire parfaitement supprimable.

(46) a. [*(*Bien que*) celle-ci soit géographiquement située en Afrique,] [l'Art de l'Égypte antique, est l'une des principales sources de l'art en Europe.]

b. [*(*Parce qu'*) il de nature susceptible,] [Paul s'énerve facilement.]

c. [(*Parce que*) Paul de nature susceptible,] [il s'énerve facilement.]

Nous avons également trouvé un exemple de cette catégorie mettant en jeu un autre problème dans nos données naturelles, nous le reproduisons en (47). Ici le connecteur utilisé, *faute de* indiquant un lien causal, n'est clairement pas supprimable sans effet sur la grammaticalité. Ce sera par ailleurs le cas pour tout connecteur de catégorie prépositionnelle. De plus, sa suppression entraîne la perte d'une information de type négation qui était contenue dans le connecteur. Ce sera également le cas pour les quelques connecteurs de polarité négative comme *de peur que*, *de peur de* ou *sans*. Même en transformant l'exemple pour le rendre grammatical, on ne pourrait pas récupérer cette information à moins de l'ajouter. Le fait d'avoir des exemples agrammaticaux dans nos données artificielles ne pose de problème qu'en regard de la modélisation des données adoptées. Il est clair que les informations syntaxiques vont ici conduire à une représentation très étrange de l'exemple. Il nous semble de toute façon que du fait même de la suppression du connecteur, nous ne pouvons nous appuyer sur une modélisation de type règles de production telle qu'utilisée dans les études précédentes. La perte de la négation pose plus de problèmes puisqu'ici, en restant sur une représentation de type paires de mots, on aurait un lien causal entre le fait de percevoir et celui de méconnaître alors que le lien s'établit entre le fait de ne pas percevoir et le fait de méconnaître. En tout, nous avons construit 730 exemples artificiels à partir de ce connecteur sur les 59 909 exemples de la relation *Explanation*[4], l'effet devrait donc être négligeable.

(47) [*(*Faute de*) percevoir les liens possibles entre la démarche linguistique et la science du mouvement et de l'équilibre des corps,] [les contemporains de Guillaume ont en effet méconnu sa quête du mouvement sous-jacent à la construction des représentations par et dans la langue.]

4.2.3 Modification de la relation inférable

La suppression du connecteur peut modifier la relation inférée entre deux segments. Cet effet rejoint les principes évoqués précédemment d'inférence préférentielle d'une relation causale et par défaut d'une continuité temporelle. Ainsi l'exemple (48a), issu de (WILSON et SPERBER, 1990), est ambigu, les deux interprétations pouvant être révélées par l'insertion des connecteurs *donc* en (48b) (*Result*) et *après tout* (*Explanation-Evidence*) en (48c). Cependant, le discours en (48a) n'est pas incohérent, une relation est inférable, le choix étant dépendant du contexte. Ces cas d'ambiguïté entraînent donc éventuellement une modification de la relation inférée lors de la suppression du connecteur.

(48) a. Pierre n'est pas stupide. Il peut trouver son chemin tout seul.

4. Nous présentons les données artificielles construites pour le français en section 4.4.1

b. Pierre n'est pas stupide. Il peut *donc* trouver son chemin tout seul.

c. Pierre n'est pas stupide. *Après tout* il peut trouver son chemin tout seul.

Concernant cet effet, SPORLEDER et LASCARIDES (2008) avaient proposé l'exemple (49) issu de leur corpus artificiel. Dans cet exemple, la suppression du connecteur *although* marquant un contraste annule l'inférence de cette relation mais le discours reste cohérent, on identifie une relation de type *Continuation*. Ici les deux relations étaient présentes à l'origine, mais la suppression du connecteur ne rend plus possible que l'inférence de la relation implicite *Continuation*.

(49) [(*Although*) the electronics industry has changed greatly,] [possibly the greatest change is that very little component level manufacture is done in this country.]

SPORLEDER et LASCARIDES (2008) n'ont pas observé cet effet pour les relations comme *Result* et *Explanation*. Cependant, nous avons trouvé dans les données ANNODIS un exemple, repris en (50), où la suppression du connecteur, ici *puisque*, semble bien modifier la relation inférée. Le connecteur marque une relation de type explication. La seconde proposition est une explication pour ce qui est énoncé dans la première : c'est parce que les Amorrites commencent à migrer qu'ils deviennent des adversaires. Ceci implique que l'évènement de migration intervient avant le fait de devenir des adversaires des souverains d'Ur. Sans le connecteur, l'ordre des évènements s'inverse et suit désormais l'ordre du texte : les Amorittes deviennent des adversaires des souverains d'Ur puis ils commencent à migrer. De plus une relation *Result* paraît inférable, le premier segment expliquant le second : comme les Amorrites deviennent des adversaires d'un ensemble d'individus, ils commencent à migrer. Ici ce n'est donc pas une relation implicite déjà présente que l'on infère en supprimant le connecteur mais une nouvelle relation. On peut expliquer cette modification par l'hypothèse d'inférence d'une continuité linéaire par défaut liée à un manque de connaissance dans le domaine. De plus, ici le contenu des arguments rend possible une interprétation causale qui suit l'ordre linéaire du texte, la cause précédant l'effet. On peut en effet accepter que le fait de devenir des adversaires, donc de risquer une potentielle lutte, encourage un peuple à migrer. On note que l'interprétation ou la portée du localisateur temporel « *alors* » est modifié par la suppression de *puisque* : avec le connecteur, il constitue une anaphore de l'expression temporelle « à la période suivante », sans, il prend un rôle discursif signalant un lien résultatif.

(50) [Les Amorrites deviennent à la période suivante de sérieux adversaires des souverains d'Ur,] [(*puisqu'*)ils commencent alors à migrer en grand nombre vers la Mésopotamie.]

Cet effet ne se retrouve pas dans tous les cas d'*Explanation* et n'est pas lié à la présence du connecteur *puisque* pour lequel nous avons donné précédemment un exemple en (42b) où cet effet n'est pas observable. La modification dépend donc ici du contenu des segments et d'un manque de connaissances dans le domaine.

La modification de la relation inférée est bien sûr un problème important pour un système automatique puisque nous allons donner au système une instance dont l'étiquette ne correspond pas à la relation que l'on peut inférer entre les arguments. Comme cet effet est plutôt rare, nous n'avons trouvé qu'un seul exemple de ce type pour *Explanation*, nous espérons que ce bruit ne sera pas très important.

4.2.4 Relations difficiles à identifier

Nous avons fait une catégorie supplémentaire pour regrouper les cas où l'inférence de la relation requiert un effort supplémentaire, ce qui correspond bien sûr à un effet difficilement quantifiable.

L'ajout de cette catégorie est légèrement biaisé par le fait que nous nous sommes intéressée essentiellement aux relations causales, donc inférées de manière préférentielle, et nous sommes aussi biaisée par le fait que nous savons qu'originellement un lien causal était explicité. Il est ainsi difficile de savoir si l'on pouvait vraiment inférer autre chose dans ces cas ou si l'on force la cohérence du discours, car comme toujours pour les phénomènes sémantiques, les locuteurs sont plutôt conciliants. Nous n'avons pas trouvé de cas où la difficulté nous amenait par exemple à identifier un contraste entre les segments. Au pire, on identifie une simple continuation ou élaboration du discours.

Cette catégorie correspond donc à des exemples où le connecteur semble redondant mais l'inférence est plus difficile. Dans l'exemple (51a), il faut ainsi se rendre compte de la proximité des chiffres 200 000 et 195 000, proximité liée au domaine de la paléontologie, pour identifier correctement le lien causal. Pour l'exemple (51b), il faut penser que si l'on parle de monnaie, alors nécessairement il y avait une cité. En (51c), l'identification semble vraiment difficile, probablement parce que nous manquons de connaissances sur le « *Ban Amendment* ». Il nous semble cependant qu'on est à la limite de l'incohérence ici. Tous ces exemples ont en commun d'appartenir à des textes relevant de domaines spécifiques pour lesquels les connaissances partagées sont limitées ce qui rend probablement l'explicitation des relations nécessaire.

(51) a. [les paléontologues donnent à Homo sapiens un âge d'environ 200 000 ans] [(*puisque*) les plus vieux ossements retrouvés sont deux crânes datés de -195 000 ans.]

b. [Des témoignages archéologiques indiquent que la cité fut créée au début de la dynastie hasmodéenne de Judée,] [(*car*) les monnaies les plus anciennes retrouvées sur le site datent du IIe siècle avant JC.]

c. [Le "Ban Amendment" a rencontré une opposition farouche parmi les groupes d'industriels et certains pays.] [(*En effet*) les États-Unis, (...) disposent d'un accord bilatéral pour exporter des déchets au Canada.]

Nous avons donc vu dans cette section que l'hypothèse de redondance du connecteur n'était pas toujours vérifiée au sens où la suppression d'un connecteur peut rendre un discours incohérent ou modifier la relation inférée mais aussi au sens où, en son absence, l'inférence de la relation est difficile car il y a peu d'indices supplémentaires. Nous pensons que ces effets ne sont cependant pas rédhibitoires mais qu'il faut les prendre en compte. Ils participent plus généralement de la différence en termes distributionnels entre données artificielles et naturelles. Dans la section suivante, nous revenons sur l'hypothèse qui suppose une similarité entre les données.

4.3 Apprentissage avec données non identiquement distribuées

Un système d'apprentissage statistique se fonde sur certaines hypothèses, en particulier les données d'entraînement et d'évaluation doivent être tirées indépendamment et identiquement de la même distribution. Cela signifie, informellement, que les données suivent une même loi de probabilité (généralement inconnue) et que leur échantillonnage a été fait de manière aléatoire. Lorsqu'un modèle est entraîné sur des données artificielles et évalué sur des données naturelles, on suppose que cette hypothèse fondamentale est respectée donc, en particulier, que ces deux ensembles de données sont identiquement distribués. Cependant on voit clairement, de par sa conception, que la méthode de création automatique d'exemples implicites proposée par MARCU et ECHIHABI (2002) risque fort de mettre à mal cette hypothèse importante. Rien en effet dans cette méthode ne garantit qu'on obtienne, pour nos données artificielles, une distribution proche de celle des données naturelles. Comme nous l'avons dit, les scores présentés par SPORLEDER et LASCARIDES (2008)

notamment montrent que l'hypothèse d'une similarité entre données naturelles et artificielles n'est pas vérifiée puisque l'ajout massif de données artificielles fait baisser les performances du système. Cette configuration correspond à un apprentissage avec des données non identiquement distribuées, un problème assez connu en apprentissage et en TAL. Dans la partie suivante, nous introduisons des notations concernant l'apprentissage statistique (section 4.3.1). Nous explicitons ensuite l'hypothèse d'identité de distribution (section 4.3.2) puis montrons que les données naturelles et artificielles présentent de nombreuses différences (section 4.3.3). Finalement, nous décrivons les différentes stratégies mises en place pour cette configuration dans le cadre de l'adaptation de domaine (section 4.3.4).

4.3.1 Apprentissage statistique

Nous nous plaçons ici dans le cadre le plus général en apprentissage supervisé, dit cadre de Vapnik ou cadre stochastique (MOHRI et al., 2012 ; RALAIVOLA, 2010). Un algorithme de classification supervisé construit une fonction f qui associe à un objet x d'un espace $\mathcal{X}$ une classe y de l'espace discret $\mathcal{Y}$ de façon adéquate, ici en minimisant l'erreur de généralisation. Pour simplifier, on se place dans le cas de la classification binaire $\mathcal{Y} = \{-1, 1\}$. On suppose que l'on dispose d'un échantillon d'exemples $S = ((x_1, y_1), \ldots, (x_m, y_m)) \in \mathcal{X} \times \mathcal{Y}$ tirés indépendamment et identiquement (i.i.d.) d'une distribution fixe inconnue D. Une paire (x_i, y_i) est la réalisation d'une variable aléatoire (X_i, Y_i) de loi D sur $\mathcal{X} \times \mathcal{Y}$. Le classifieur utilise S pour sélectionner la fonction d'hypothèse $f \in H$ où H est l'ensemble des fonctions d'hypothèse possibles ou classe d'hypothèses. Le but de l'apprentissage est de trouver la fonction d'hypothèse qui est la meilleure approximation de la fonction d'étiquetage cible, celle qui correspond à l'erreur de généralisation la plus faible par rapport au concept cible. L'erreur de généralisation, ou risque réel, pour une hypothèse $f \in H$ est définie par la fonction de risque $R(f)$ en fonction de D et d'une fonction de perte $l : \mathcal{Y} \times \mathcal{Y}' \to \mathbb{R}_+$ qui mesure la différence, ou perte, entre l'étiquette prédite et l'étiquette réelle.

$$R_l(f) = \mathbb{E}[l(f(X), Y)] = \int_{\mathcal{X}\times\mathcal{Y}} l(f(x), y) dD(x, y)$$

Dans le cas de la perte zéro-un définie par :

$$l(f(x), y) = \begin{cases} 1 \text{ si } f(x) \neq y \\ 0 \text{ sinon} \end{cases}$$

le risque correspond à la probabilité que la variable aléatoire $(X, f(X))$ soit différente de (X, Y) pour $(X, Y) \sim D$:

$$R_l(f) = Pr_{(X,Y)\sim D}[f(x) \neq y]$$

En pratique, la distribution $P(X, Y)$ est inconnue, seul l'échantillon S nous fournit des informations sur la distribution. L'erreur de généralisation de l'hypothèse n'est pas directement accessible. On se fonde donc sur l'erreur empirique $\hat{R}(f)$ pour déterminer l'adéquation de la fonction d'hypothèse f_S selon la distribution empirique $\widetilde{P}(X, Y)$ des données :

$$\widehat{R}_l(f_S) = \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \widetilde{P}(x,y)l(f(x),y) = \frac{1}{m}\sum_{i=1}^{m} l(f_S(x_i),y_i)$$

Le but de l'apprentissage est donc de minimiser ce risque empirique à partir des données :

$$\widehat{f}_S = \text{argmin}_{f\in H}\widehat{R}(f)$$

4.3.2 L'hypothèse d'identité de distribution

Selon l'hypothèse i.i.d., chaque $Z_i = (X_i, Y_i) \in \mathcal{X}\times\mathcal{Y}$ est un tirage indépendant d'un même modèle probabiliste fixé. L'indépendance signifie que la probabilité d'observer Z_i et Z_j est simplement le produit des probabilités d'observer chacune des variables. Même si l'hypothèse d'indépendance est probablement également problématique dans notre cas comme pour de nombreuses tâches de TAL, nous nous intéressons ici seulement à l'hypothèse d'identité qui pose problème dans la configuration correspondant à l'utilisation d'exemples explicites/artificiels et implicites.

La distribution d'une variable aléatoire Z est l'ensemble des probabilités que sa valeur appartienne à des sous-ensembles de son ensemble de définition. Dire que deux variables aléatoires Z_i et Z_j ont la même distribution ou sont identiquement distribuées correspond au fait que $Pr[Z_i \in A] = Pr[Z_j \in A]$ pour tout sous-ensemble $A \subset \mathcal{X}\times\mathcal{Y}$. En particulier, ces variables doivent avoir la même distribution pour les sous-ensembles d'entraînement et d'évaluation. Cette hypothèse n'est pas toujours vérifiée dans une configuration réaliste, c'est une hypothèse simplificatrice qui permet notamment de comprendre les propriétés des procédures d'apprentissage (capacité de généralisation) et de construire de nouveaux algorithmes.

Cette hypothèse peut être décomposée en différents cas qui ont reçu des noms et des solutions différentes. Nous reprenons ici les catégories décrites en adaptation de domaine, par exemple dans (Moreno-Torres et al., 2012 ; Jiang, 2008). Cette catégorisation découle de la réécriture de la probabilité jointe d'observer en même temps les évènements $X = x$ et $Y = y$ en utilisant les probabilités marginales et conditionnelles :

$$Pr((X = x, Y = y)) = Pr(Y = y|X = x)Pr(X = x) = Pr(X = x|Y = y)Pr(Y = y)$$

Le premier cas étudié correspond à une différence au niveau de la distribution des variables de sortie $Pr(Y = y)$, c'est un problème en particulier pour les modèles génératifs d'où l'appellation de biais dans les probabilités prior (*prior probability shift*). On suppose ici que $Pr(X = x|Y = y)$ reste inchangée. Ce cas est similaire au problème de déséquilibre des classes présenté dans le chapitre précédent qui est considéré comme un biais délibéré dans les données, résultant de la conception du problème, par exemple parce qu'il serait trop coûteux d'obtenir plus d'observations pour les classes rares.

Le second cas, le plus étudié, correspond à une différence au niveau des variables d'entrée $Pr(X = x)$. Lorsque les données disponibles à l'entraînement et à l'évaluation ne suivent pas la même distribution marginale sur les entrées, on parle de *covariate shift*, une variation ou un biais dans les observations. On peut trouver également dans la littérature le terme de *population drift*, il s'applique cependant plutôt au cas où la distribution de la population, les variables en entrée, change dans le temps. Les nombreux travaux cherchant à gérer ce problème supposent que $Pr(Y = y|X = x)$

reste inchangée. À première vue, ce problème n'en est donc pas un, puisque ce que l'on cherche à modéliser c'est justement cette probabilité conditionnelle. Cependant, il a été montré que l'on pouvait obtenir des améliorations en prenant en compte ce biais dans le cas où la classe de modèles considérée ne contient pas la vraie distribution conditionnelle, le modèle optimal sélectionné dépend alors de $Pr(X)$. Intuitivement, le modèle optimal a de meilleures performances dans les régions denses de $\mathcal{X}$ que dans les régions non denses donc si ces régions denses ne sont pas les mêmes dans les deux ensembles de données, alors ce modèle ne sera plus optimal sur les nouvelles données.

Enfin, le dernier cas envisagé dans la littérature concerne une différence sur la distribution conditionnelle $Pr(Y = y|X = x)$ en supposant que $Pr(X = x)$ reste inchangée. On parle alors de *concept shift* ou *concept drift*, c'est-à-dire une modification de la fonction cible ou du concept que l'on cherche à modéliser.

Les hypothèses faites à chaque fois, laissant inchangée une partie de la distribution jointe, simplifient la compréhension du problème et, selon MORENO-TORRES et al. (2012), les cas où plusieurs distributions sont différentes simultanément sont rares et très difficiles à gérer. Cependant, les solutions mises en place dans la littérature ne rendent pas toujours clairement compte du type de biais qui est géré et notamment les deux derniers cas sont relativement mal séparés (SØGAARD, 2013). Par exemple, l'apprentissage multi-tâche est une solution envisageable mais il n'est pas évident de savoir si cette méthode agit sur la représentation ou sur l'hypothèse construite. Concernant le second cas, le biais dans les observations, plutôt que de supposer que $Pr(Y = y|X = x)$ reste inchangée, on peut faire l'hypothèse que sous une certaine modification de l'espace d'entrée $P(X)$, il existe une fonction cible qui a de bonnes performances sur les ensembles de données non i.i.d. considérés. Concernant le troisième cas, relativement peu étudié, les solutions consistent généralement en une sélection d'instances, donc on agit en fait au niveau de l'espace d'entrée également.

L'apprentissage avec des données de distributions différentes a donné lieu à des solutions regroupées sous des appellations variées. On parle en général d'apprentissage par transfert en apprentissage et plus souvent d'adaptation de domaine en TAL bien que les deux termes ne semblent pas totalement se recouvrir. On peut également parler de manière très générale d'apprentissage sous biais (SØGAARD, 2013). Le terme de domaine fait originellement référence à la question du genre des textes mais il a pu être étendu à des différences plus restreintes ou plus génériques comme la période temporelle de production des textes, des variations dans des flux de données variant dans le temps ou la gestion du bruit dans les données. Ce problème est en fait assez commun en TAL et a fait l'objet notamment ces dernières années de nombreuses études, de *workshop* dédié comme lors de la conférence ACL en 2010 (*Domain Adaptation for Natural Language Processing*) ou pour le cadre de la traduction automatique en 2012 (*Domain Adaptation in Machine Translation*) et de tâches partagées pour l'analyse syntaxique automatique lors de la conférence CoNLL. Après avoir détaillé dans la partie suivante les caractéristiques de nos deux ensembles de données qui engendrent des différences en termes distributionnels, nous présentons brièvement les différentes solutions envisagées dans la littérature.

4.3.3 Différences de distribution entre données naturelles et artificielles

Pour rappel, notre problème consiste à utiliser des données implicites artificielles, construites à partir d'exemples explicites, pour apprendre une fonction de classification adéquate sur des données implicites naturelles. Les données naturelles et artificielles ont le même espace de sortie, des étiquettes de relation, et des entrées similaires, des paires de segments textuels. Ces deux

ensembles de données sont néanmoins distribués différemment, et ce, à plusieurs titres. D'une part, les données artificielles sont par définition obtenues à partir d'exemples de relations explicites : il n'y a aucune garantie que ces données soient distribuées comme les exemples de relations implicites attestés. La différence porte tant sur la distribution des étiquettes (les relations) que sur l'association entre étiquettes et entrées (les paires des segments) à classer. En outre, la suppression du connecteur modifie les exemples, ce qui peut avoir une incidence comme noté dans la section précédente. Enfin, les données artificielles sont généralement obtenues automatiquement à partir d'heuristiques et/ou de modèles [5] ce qui induit un bruit potentiel à la fois dans l'étiquetage en relation et dans la segmentation que l'on ne retrouve pas dans les données naturelles. Nous décrivons dans la suite de cette section les différences entre les deux types de données qui induisent des biais entre les distributions marginales et conditionnelles.

4.3.3.1 Déséquilibre des classes (*class imbalance* ou *prior probability drift*)

Dans le cas de biais au niveau de l'espace de sortie, la différence porte sur la distribution marginale des classes. Les distributions marginales des classes sont déjà différentes entre données explicites et implicites naturelles, et les heuristiques utilisées pour créer les données artificielles en produisent une nouvelle. Les chiffres exacts sont donnés dans la section 4.4.1 pour les données françaises et dans la section 4.4.2 pour les données anglaises.

Pour le français, on peut noter que la classe sous-représentée dans les données naturelles, *Contrast*, devient sur-représentée dans les données artificielles. Pour cette classe, la forme *mais*, toujours en emploi discursif, a permis d'extraire 75 % des données. En revanche, la relation *Continuation* devient sous-représentée dans les données artificielles pour le français. Les connecteurs de cette relation, comme *et*, sont plus ambigus en emploi et nous avons dû définir des motifs plus stricts pour éviter de récupérer de mauvais exemples.

Pour l'anglais, nous construisons un corpus artificiel à partir des données explicites naturelles et un autre à partir de données brutes. Pour le premier, on conserve donc la distribution des données explicites qui est différente de celle des données implicites. Par exemple, au niveau 1 de relation, la classe *Temporal* ne représente que $5,1$% des données implicites (voir section 3.4.1, tableau 3.4) contre $18,6$% des données explicites (voir section 4.4.2.1, tableau 4.5). On a également d'importantes différences au niveau 2 de relation. Par exemple nous avons dit dans le chapitre précédent que nous ne disposions pas de suffisamment d'exemples de relation implicite pour *Condition*, seul 1 exemple est annoté. Dans les données explicites, cette relation correspond à $1\,204$ exemples. Au contraire, nous disposons de 69 annotations de la relation *Pragmatic Cause* dans les données implicites contre seulement 8 dans les données explicites. Pour les données artificielles construites automatiquement, la distribution est relativement proche des données explicites naturelles (cf. section 4.4.2.1) puisque nous avons utilisé des modèles construits sur ces dernières plutôt que des heuristiques pour étiqueter les données. On a cependant des différences, bien que moins importantes qu'entre implicites et explicites, qui peuvent provenir d'erreurs des modèles mais aussi probablement de la différence en termes de domaine entre les corpus utilisés : les données brutes sont également constituées d'articles journalistiques mais de sources plus variées avec aussi des dates de production différentes.

Ce biais peut être facilement géré en ré-échantillonnant les données artificielles suivant la distribution des données implicites naturelles.

5. Nous décrivons la construction des données artificielles dans la section 4.4.

4.3.3.2 Biais dans les observations (*covariate shift* ou *population drift*)

Le problème de biais dans les données correspond à une différence portant sur la distribution marginale des entrées, les paires de segments. Le fait d'utiliser des exemples explicites induit une différence. On peut en effet penser que sans connecteur les indices utilisés sont différents. De plus, la suppression du connecteur peut aboutir à des exemples agrammaticaux ou incohérents comme nous avons pu le voir dans la section précédente. La segmentation induit aussi des différences. Dans le cas des données artificielles obtenues automatiquement, on a d'une part potentiellement des erreurs de segmentation dans les données artificielles. D'autre part, la segmentation des données artificielles correspond à des hypothèses simplificatrices : un argument couvre au plus une phrase et on a au plus deux arguments par phrase. La segmentation des données naturelles ne suit bien sûr pas ces hypothèses : les arguments peuvent être aussi multi-phrastiques ou séparer une phrase en plus de deux segments du moins dans le corpus français ANNODIS. Rappelons que dans les données implicites du PDTB, les relations s'établissent entre des phrases adjacentes ou des propositions séparées par deux points ou un point virgule, avec éventuellement un dépassement de la phrase ou, au contraire, la supression de certains segments (principe de minimalité). Ceci correspond à des hypothèses simplificatrices dans l'annotation des implicites, hypothèses qui correspondent seulement en partie à celles utilisées pour construire automatiquement des données artificielles, mais qui ne correspondent pas du tout à la segmentation des données explicites naturelles. Enfin, on a potentiellement un biais en termes de genre : pour le français, les exemples artificiels sont tous construits à partir de l'*Est Républicain* mais les données naturelles proviennent aussi de Wikipédia ; pour l'anglais, les données naturelles et artificielles peuvent provenir du même corpus ou du *Bllip*, un autre ensemble d'articles journalistiques.

4.3.3.3 Modification de la fonction cible (*concept drift* ou *functional relation change*) ou du processus de génération des observations

Nous l'avons dit, le troisième cas concerne la distribution conditionnelle $Pr(Y = y|X = x)$. Cette distribution est biaisée dans notre configuration puisqu'il est possible d'avoir un exemple artificiel similaire à un exemple naturel mais portant une étiquette différente. C'est le cas lorsque la suppression du connecteur entraîne une modification de la relation inférée, l'effet vient donc de la méthode mais aussi de la différence entre explicite et implicite, l'explicitation d'un exemple permettant de guider l'interprétation. Cette situation peut aussi être due à des erreurs d'étiquetage lors de l'annotation automatique des exemples artificiels : soit il n'y a en fait aucune relation entre les arguments, soit une autre relation était présente.

Le problème est cependant plus large, nous avons aussi des différences au niveau de la distribution conditionnelle $Pr(X = x|Y = y)$. Nous n'avons pas trouvé de terme pour cette situation, on peut parler de biais dans la génération des observations. On peut se rendre compte de la différence de distribution sur l'association entre étiquettes et exemples en considérant certaines caractéristiques des données. Pour le français, on peut par exemple regarder la répartition entre occurrences de relations inter- et intra-phrastiques (la relation s'établit entre deux phrases ou deux segments à l'intérieur d'une phrase), voir tableau 4.1. Entre occurrences de relations implicites naturelles et artificielles, on a une proportion d'inter-phrastiques similaire pour *Contrast* ($57{,}1$ % d'inter-phrastiques dans les deux types de données), proche pour *résultat* ($45{,}7$ % d'inter-phrastiques dans les données naturelles, $39{,}8$ % dans les artificielles) mais très différente pour *continuation* ($70{,}0$ % d'inter-phrastiques dans les naturelles, $96{,}5$ % dans les artificielles), et pour *explication* ($21{,}4$ % dans les naturelles, $53{,}0$ % dans les artificielles). On observe aussi que la proportion d'occurrences de relations inter-phrastiques dans les données artificielles ne reflète pas celle des données explicites, ceci étant dû à notre heuristique.

Relations	Implicites	Explicites	Artificiels
Contrast	57,1 % (57,1 %)	40,0 %	57,1 %
Result	50,9 % (45,7 %)	65,4 %	39,8 %
Continuation	66,9 % (70,0 %)	52,5 %	96,5 %
Explanation	21,4 % (21,4 %)	37,9 %	53,0 %
Total	494 (252)	614	392 260

Table 4.1.: Corpus ANNODIS : répartition des occurrences de relations inter-phrastiques implicites (naturelles), explicites et artificielles pour tous les exemples disponibles pour les données françaises, (X %) pour les seuls exemples utilisés dans nos expériences dans le cas des implicites.

4.3.4 Adaptation de domaine

L'adaptation de domaine correspond à la situation générale où l'on veut apprendre un modèle performant sur des données d'un domaine dit *cible* à partir de données ou d'un modèle d'un domaine dit *source*. La différence en termes de domaine correspond à des différences en termes distributionnels du type de celles décrites précédemment. L'adaptation dite multi-domaine s'intéresse aux cas où l'on dispose de données provenant de plus de deux domaines, nous restons ici dans le cadre où l'on a deux domaines. L'hypothèse fondamentale qui sous-tend ces travaux est que les domaines bien que différents sont liés. Si les domaines sont trop différents, on ne pourra rien apprendre, rien transférer entre eux. La question de savoir quand on peut supposer que des domaines sont trop différents n'est cependant pas résolue. LI (2012) estime que les méthodes reposant sur une pondération d'instances sont adaptées aux cas où l'écart entre les domaines est petit tandis que les autres méthodes, fondées sur les traits ou les modèles, permettent de s'adapter à des écarts plus importants. Nous n'avons cependant pas trouvé de mesure permettant d'évaluer si la distance entre les domaines est « trop » grande. On peut bien sûr utiliser des mesures de divergence entre distributions ou des mesures plus dépendantes de la tâche, comme le nombre de mots inconnus, mais il n'existe pas de valeur seuil minimale permettant d'affirmer que l'on a affaire à un cas où il est nécessaire d'effectuer une forme d'adaptation non plus que de valeur maximale au-delà de laquelle les données seraient trop différentes pour quelque transfert que ce soit. Il semble que la seule réponse réside dans des tests empiriques : les expériences de SPORLEDER et LASCARIDES (2008) notamment montrent, de par la baisse de performance observée, que l'on a une différence donc que l'on pourrait avoir besoin d'une forme d'adaptation tandis que les expériences de WANG et al. (2012) et RUTHERFORD et XUE (2015) montrent, de par leur succès, que les données ne sont pas trop différentes donc qu'une adaptation est possible.

On peut voir l'adaptation de domaine comme un cas particulier d'apprentissage par transfert entre deux tâches, au sens où l'on veut transférer les connaissances que l'on a sur le domaine source au domaine cible. Cependant, l'apprentissage par transfert est généralement supervisé, on a des données étiquetées pour les deux tâches. De plus, les deux tâches sont différentes, elles n'ont pas forcément le même jeu d'étiquettes, et l'on apprend les tâches indépendamment. Quand les tâches sont apprises simultanément, on parle d'apprentissage multi-tâche. Quand il y a une seule tâche et un seul jeu d'étiquette, on parle d'adaptation de domaine. De plus, l'adaptation de domaine peut être supervisée, semi-supervisée ou non supervisée. On dispose toujours, dans ce cadre, de données ou d'un modèle sur le domaine source. Le degré de supervision dépend de l'information disponible dans le domaine cible. On parle d'adaptation supervisée quand on dispose de données annotées cibles étiquetées, d'adaptation semi-supervisée quand on dispose d'un petit ensemble de données cibles étiquetées et d'un grand ensemble de données cibles non étiquetées et d'adaptation non supervisée quand on ne dispose que d'un grand ensemble de données cibles non étiquetées. Les données cibles étiquetées sont toujours en faible quantité, sinon l'adaptation n'est pas nécessaire. On peut utiliser des méthodes d'apprentissage multi-tâche pour faire de l'adaptation de domaine supervisée, en considérant chaque domaine comme correspondant à une tâche différente. C'est par

ailleurs la méthode adoptée par LAN et al. (2013) pour l'identification des relations implicites à partir d'exemples explicites bien que ces auteurs ne justifient pas directement l'utilisation de cette stratégie par la différence en termes distributionnels. On peut également se placer dans le cadre de l'apprentissage semi-supervisé pour faire de l'adaptation de domaine non supervisée afin de prendre en compte la différence en termes de distribution marginale sur les entrées.

Les méthodes d'adaptation de domaine ont été utilisées pour différentes tâches de TAL comme l'étiquetage morpho-syntaxique (JIANG et ZHAI, 2007 ; BEN-DAVID et al., 2007 ; DAUMÉ III, 2007 ; DAUMÉ III et al., 2010), l'analyse syntaxique (MCCLOSKY et al., 2010), la reconnaissance et/ou le typage d'entités nommées (DAUMÉ III et MARCU, 2006 ; JIANG et ZHAI, 2007 ; FINKEL et MANNING, 2009), la recapitalisation de texte (CHELBA et ACERO, 2004 ; DAUMÉ III et MARCU, 2006), la classification de documents (PETASIS, 2011 ; LONG et al., 2012), la traduction automatique (KOEHN et SCHROEDER, 2007 ; AXELROD et al., 2011), le filtrage de spam (JIANG et ZHAI, 2007) ou la désambiguïsation de mots (CHAN et NG, 2005).

La différence distributionnelle entre données d'entraînement et d'évaluation est un problème assez général : lorsque l'on collecte des données, elles comportent toujours un biais, elles sont tirées d'une distribution spécifique parce que l'on s'est restreint à un certain genre, à une période temporelle, à un guide d'annotation, etc. . . En apprenant un modèle sur ces données, on effectue donc un apprentissage sous biais, on ne modélise que la distribution des données disponibles. Or on veut généralement un modèle généralisable, c'est-à-dire permettant d'obtenir de bonnes performances sur de nouvelles données. Une première solution envisageable est donc de construire un modèle moins attaché aux données d'entraînement, un modèle qui, en quelque sorte, ne modélise que les informations génériques de la tâche, communes aux deux domaines. Cette solution est généralement trop simpliste mais l'idée générale de transférer une information limitée entre les ensembles de données a fondé la plupart des méthodes de ce cadre. Nous présentons dans cette section les différentes méthodes proposées en adaptation de domaine qui se divisent en deux grandes catégories selon que l'on agit sur l'espace des entrées ou sur l'espace des modèles. De bonnes synthèses sur ces différentes méthodes peuvent être trouvées dans (JIANG, 2008 ; LI, 2012 ; SØGAARD, 2013). Nous laissons de côté ici le problème de biais dans les probabilités prior, au sens des probabilités marginales sur les sorties, qui sera plus généralement vu comme un cas de déséquilibre de classe pour lequel on utilisera le fait que, dans notre cas, on connaît la distribution des classes dans les données cibles, c'est-à-dire nos données naturelles. On pourra donc utiliser des méthodes de rééchantillonnage ou de pondération des instances pour obtenir une distribution similaire entre les deux ensembles de données. Notons que dans le cadre d'un modèle génératif, ce cas correspond à une modification au niveau de l'espace des modèles puisque les probabilités prior entrent directement dans la modélisation du problème.

4.3.4.1 Espace des entrées

Les méthodes agissant au niveau des entrées reposent soit sur une pondération des instances soit sur une modification de leur représentation.

Une façon d'envisager le problème de l'adaptation de domaine est de considérer que certaines instances du domaine source n'ont aucune utilité voire sont néfastes : elles sont trop spécifiques à ce domaine et risquent de corrompre le modèle. On peut donc chercher à les supprimer. De manière moins radicale, on peut aussi envisager que certaines instances sont plus pertinentes que d'autres pour adapter un système à un nouveau domaine : on peut donc pondérer les instances selon leur pertinence, la sélection étant un cas particulier de pondération. Cette idée n'est pas spécifique à l'adaptation de domaine, elle fonde aussi des stratégies permettant de diminuer le coût computationnel ou le bruit pour certaines méthodes en apprentissage supervisé ou semi-

supervisé. Ce qui change, c'est l'information disponible, l'accès ou non à des données annotées cibles, et éventuellement le critère de pertinence, fondé uniquement sur les données cibles et non sur l'ensemble des données.

Une large proportion d'études dans ce cadre correspond à l'estimation du quotient $\frac{P_T(X)}{P_S(X)}$ avec $P_T(X)$ la distribution marginale sur les entrées du domaine cible et $P_S(X)$ celle pour le domaine source. En effet, étant donné que l'on s'intéresse ici à un modèle performant sur les données cibles, la définition du modèle optimal $\widehat{f}$ dans le cadre de la minimisation du risque réel peut se réécrire en fonction de la distribution sur les données cibles $P_T(X,Y)$:

$$\widehat{f_S} = \text{argmin}_{f \in H} \sum_{(x,y) \in X \times Y} l(f(x), y) P_T(x, y)$$

Étant donné que l'on dispose de données issues de la distribution source, on va vouloir exprimer cette erreur en tenant aussi compte de $P_S(X,Y)$:

$$\widehat{f_S} = \text{argmin}_{f \in H} \sum_{(x,y) \in X \times Y} l(f(x), y) \frac{P_T(x, y)}{P_S(x, y)} P_S(x, y)$$

La distribution jointe réelle du domaine source n'est pas connue, mais on peut utiliser une approximation de cette distribution sur l'échantillon utilisé pour l'apprentissage $\widetilde{P}_S(x,y)$, le risque empirique se réécrit donc, avec m la taille de l'échantillon source :

$$\begin{aligned}\widehat{f_S} &= \text{argmin}_{f \in H} \sum_{(x,y) \in X \times Y} l(f(x), y) \frac{P_T(x, y)}{P_S(x, y)} \widetilde{P}_S(x, y) \\ &= \text{argmin}_{f \in H} \frac{1}{m} \sum_{i=1}^{m} \frac{P_T(x_i^S, y_i^S)}{P_S(x_i^S, y_i^S)} l(f(x_i^S), y_i^S)\end{aligned}$$

Si l'on fait l'hypothèse que $P_T(Y = y|X = x) = P_S(Y = y|X = x)$, on obtient donc la formule suivante :

$$\widehat{f_S} = \text{argmin}_{f \in H} \frac{1}{m} \sum_{i=1}^{m} \frac{P_T(x_i^S)}{P_S(x_i^S)} l(f(x_i^S), y_i^S)$$

Cette équation fournit une solution au problème de biais dans les observations généralement appelé pondération par importance (*importance weighting*). C'est la meilleure fonction de poids pour ce problème, ce qu'a montré (SHIMODAIRA, 2000) dans le cadre d'un biais dit de sélection, biais considéré en statistique pour refléter le problème de la sélection d'un échantillon de données observées non représentatif de la population entière. Il s'agit donc de sous-pondérer les instances observées (source) qui ne sont pas importantes pour une estimation correcte par rapport à la population (cible). On ne peut cependant pas en général calculer directement les probabilités marginales des données. Les solutions apportées à ce problème consistent donc en une estimation de ces densités. SHIMODAIRA (2000) et MASASHI et KLAUS-ROBERT (2005) proposent d'estimer directement les densités pour chaque domaine en utilisant des estimateurs non paramétriques par noyau (ou méthode de Parzen-Rozenblatt), méthode cependant impossible à utiliser dans le cas de données de haute dimensionnalité. D'autres études ont proposé de transformer l'estimation de ce ratio en la modélisation du problème d'appartenance à l'un ou l'autre des domaines (ZADROZNY,

2004 ; BICKEL et SCHEFFER, 2007 ; BICKEL et al., 2007). Intuitivement, le ratio indique combien de fois une instance devrait être présente dans l'ensemble source s'il était gouverné par la distribution cible. HUANG et al. (2007) utilisent des données cibles non étiquetées pour apprendre une fonction de poids qui minimisent la différence entre les moyennes sur les données dans un espace induit par un certain noyau (algorithme de *Kernel Mean Matching*), méthode qui nécessite un nombre suffisant de données cibles. D'autres formes de pondération ont été proposées, par exemple SØGAARD (2011) met en place, dans le cadre de l'adaptation d'analyseurs syntaxiques en dépendance, une phase de sélection parmi les exemples sources afin de ne conserver que les exemples les plus similaires aux exemples cibles, la similarité étant fondée sur la perplexité d'un modèle de langue, et SUGIYAMA et al. (2008) une procédure fondée sur une minimisation de la divergence Kullback-Leibler. Enfin, DAI et al. (2007) utilisent un algorithme de boosting qui optimise un poids pour chaque instance lié à l'erreur d'un modèle entraîné sur des données cibles et sources : le poids augmente pour une instance cible mal classée et diminue pour une instance source mal classée.

On peut également citer dans ce cadre des méthodes utilisées notamment en apprentissage semi-supervisé ayant pour visée de diminuer le nombre d'instances pour réduire le coût computationnel. Ainsi, DAELEMANS et al. (1999) testent la stratégie de condensation, qui consiste à supprimer les points éloignés des frontières donc qui n'ont pas vraiment d'influence sur leur définition, et la stratégie d'édition, qui correspond au contraire à supprimer des points proches des frontières afin de les simplifier et de rendre le classifieur moins sensible aux données déviantes et au bruit. Ces stratégies reposent sur des critères de typicalité, correspondant à une similarité inter- et intra-classe, et de force prédictive, une instance ayant une faible force prédictive est un mauvais prédicteur pour les points de son voisinage. On peut envisager d'utiliser ces méthodes pour sélectionner les instances du domaine source qui sont trop exceptionnelles par rapport aux données cibles, soit parce qu'elles sont atypiques soit parce qu'elles portent une étiquette qui ne correspond pas à celle de ses voisins. De la même manière, on peut envisager d'utiliser les méthodes d'identification des données déviantes (*outlier detection*). Ces solutions ressemblent à ce qui a été fait pour notre tâche par WANG et al. (2012).

Quand on pondère des instances, on modifie l'espace de représentation puisque l'estimation des paramètres se fait différemment. Au lieu de sélectionner des instances, on peut imaginer sélectionner ou pondérer directement les dimensions. Cette solution semble plus fine puisque l'on conserve toutes les instances avec l'hypothèse que toute instance, quel que soit son domaine d'origine, peut fournir une information utilisable pour apprendre une bonne fonction de classification pour le domaine cible. L'analyse du problème de l'adaptation de domaine dans (BEN-DAVID et al., 2007) montre qu'un système qui conduit à de bonnes performances en adaptation de domaine est corrélé à une diminution de la distance entre les données. Une autre façon d'envisager le problème est de faire l'hypothèse que, étant donné que les domaines sont proches, il existe probablement un espace dans lequel ils sont similaires. Plus formellement, même si $P_T(Y|X) \neq P_S(Y|X)$ on peut faire l'hypothèse que sous un certain changement de représentation Φ on a $P_T(y|\Phi(x)) \sim P_S(y|\Phi(x))$. L'une des solutions proposées correspond à considérer que les traits représentant les données peuvent être soit spécifiques à l'un des domaines soit génériques.

DAUMÉ III (2007) propose d'augmenter l'espace des traits en faisant pour chaque trait une copie spécifique au domaine de l'instance et une copie générique. Par exemple, dans le cas de l'étiquetage morpho-syntaxique, le trait indicateur de déterminant pour le mot « *the* » devrait correspondre à un paramètre de valeur élevée pour la version générique. Au contraire, « *monitor* » aura une version spécifique en tant que nom plus pertinente dans des manuels informatiques et en tant que verbe pour des articles de journaux. Un problème de cette méthode est qu'elle ne peut pas gérer le cas où les traits spécifiques à un domaine sont différents mais ont des corrélations cachées, contrairement à la stratégie proposée dans (BLITZER et al., 2006). De plus, cette approche ne

fait ni distinction ni connexion entre les termes de régularisation entre les traits spécifiques et indépendants des domaines. Cette approche est finalement un cas particulier d'une méthode basée sur les prior, avec différentes prior pour des traits spécifiques et indépendants des domaines (FINKEL et MANNING, 2009). Ici, le terme *prior* est à comprendre dans le cadre d'un modèle discriminant : pour ce type de modèle, on utilise généralement une prior gaussienne pour régulariser le modèle, les paramètres à apprendre sont gouvernés par cette distribution prior. Ajuster cette distribution prior peut aider dans le cadre de l'adaptation de domaine à produire une fonction d'hypothèse adéquate. Nous décrivons le modèle de FINKEL et MANNING (2009) dans la partie suivante car il est plutôt fondé sur une action au niveau du classifieur. BLITZER et al. (2006) proposent quant à eux de construire une projection de la représentation originale vers un espace commun de plus basse dimensionalité. Pour cela, ils se fondent sur de l'apprentissage multi-tâche en construisant des prédicteurs pour chaque trait d'un sous-ensemble de traits dits pivots qui ont un comportement similaire à travers les domaines. Les vecteurs de paramètres associés à ces prédicteurs encodent la covariance des traits non pivots par rapport aux traits pivots. Ils utilisent ensuite l'algorithme de décomposition en valeurs singulières (*Singular Value Decomposition*) pour compresser la matrice des paramètres des prédicteurs qui est utilisée pour projeter les données vers une représentation partagée. L'apprentissage et la prédiction se fondent sur les traits originels et les nouveaux traits. Cette méthode permet d'estimer une correspondance entre des traits qui seraient complètement séparés dans la méthode de DAUMÉ III (2007). Il faut cependant définir l'ensemble de traits pivots, les auteurs ne précisant pas la dépendance entre les performances et le choix de cet ensemble.

Une dernière lignée de travaux se fonde sur des modèles génératifs probabilistes dits modèles de sujet (*topic model*) qui permettent de découvrir et modéliser des sujets abstraits ou cachés dans des textes non étiquetés en se fondant sur une représentation distributionnelle. Différents types de contextes sont corrélés à travers les distributions des thèmes latents. Par exemple, le modèle appelé Allocation de Dirichlet Latente (LDA) suppose qu'un document est une mixture de thèmes dans lequel chaque mot est attribuable à un thème. Dans le cadre de l'adaptation de domaine, apprendre les thèmes latents à partir de plusieurs domaines peut permettre de réduire l'écart entre les domaines. Ainsi, deux termes peuvent n'apparaître chacun que dans un domaine mais ils peuvent apparaître dans les mêmes contextes entre les domaines ce qui révèle leur similarité. GUO et al. (2009) utilisent un modèle type LDA pour adapter un système dans le cadre de la reconnaissance d'entités nommées. Le modèle de sujet permet de construire des traits à partir d'un grand ensemble de données non annotées provenant des domaines sources et cibles qui sont ensuite utilisés pour entraîner un modèle sur le domaine source. Ces traits permettent de lier des informations contextuelles à travers les domaines. Cependant, le fait que la modélisation s'effectue à partir des deux domaines implique que les contextes dans les deux domaines sont générés à partir de la même distribution alors que la distribution des thèmes est probablement différente. Dans le cadre de la classification de documents, XUE et al. (2008) construisent un modèle de sujet par domaine, partageant une même distribution des termes par rapport aux thèmes, qui sont ensuite combinés. Ceci permet de rendre compte du fait que les domaines partagent des thèmes cachés communs. L'hypothèse d'un partage de la distribution termes-thèmes est cependant assez forte, elle peut différer entre les domaines. La méthode se focalise sur la distribution des thèmes par rapport aux documents, ce qui la rend peu adaptable à d'autres tâches.

4.3.4.2 Espace des classifieurs

Au lieu de pondérer les instances ou d'en modifier la représentation, certaines méthodes agissent directement au niveau des classifieurs utilisés. FLORIAN et al. (2004) ont proposé une solution simple qui consiste à ajouter des traits correspondant aux prédictions de systèmes entraînés sur différents jeux de données, ici potentiellement avec différents jeux d'étiquettes. Ce principe est

repris dans (DAUMÉ III, 2007) et correspond au meilleur système de référence. Parmi ces systèmes de référence, DAUMÉ III (2007) utilise également une interpolation linéaire des deux modèles pré-entraînés sur chaque domaine séparément, le poids de chaque modèle étant géré par un paramètre.

Cette dernière solution ne lie pas l'apprentissage des paramètres entre les deux domaines. Au contraire, CHELBA et ACERO (2004) construisent un modèle par maximum d'entropie sur les données cibles étiquetées disponibles mais régularisé par un premier modèle appris sur les données sources. Ainsi, les poids sur les paramètres peu représentés ou absents des données cibles reçoivent une valeur apprise sur les données sources tandis que les paramètres spécifiques au domaine cible sont mis à jour normalement au cours de l'apprentissage. Le modèle final est donc proche des données sources à moins que les données cibles ne le forcent à s'en éloigner. Ici les deux modèles sont appris indépendamment. FINKEL et MANNING (2009) proposent une solution similaire mais à travers un modèle joint. Dans cette étude, les auteurs répliquent les traits en une version spécifique et générique comme DAUMÉ III (2007) mais chaque ensemble correspond à un vecteur de paramètres et à un terme de régularisation séparés qui s'inscrivent dans une hiérarchie permettant une influence entre les versions génériques et spécifiques. En haut de la hiérarchie, les paramètres génériques sont associés à une prior gaussienne de moyenne nulle, ces paramètres sont utilisés comme moyenne de la prior pour les paramètres spécifiques à chaque domaine. Ces paramètres spécifiques sont ensuite utilisés pour définir une fonction de vraisemblance pour chaque domaine, la fonction objective finale correspondant à une somme sur la fonction de vraisemblance sur chaque domaine associé à son propre terme de régularisation. Ainsi, les paramètres génériques influencent les paramètres spécifiques et, inversement, les paramètres spécifiques affectent conjointement les paramètres génériques, ce qui fait que les paramètres génériques dominent les dimensions à moins que les informations apportées par les données des différents domaines ne l'emportent. Enfin, DAUMÉ III et MARCU (2006) et STORKEY et SUGIYAMA (2007) utilisent des modèles de mélange, le premier utilisant un modèle de mélange à trois composants, un spécifique à chacun des deux domaines et un générique, les seconds un modèle de mélange plus général, les domaines source et cible pouvant partager plus d'un composant de mélange.

4.4 Corpus artificiels

Nous décrivons dans cette section les corpus artificiels constitués pour le français (section 4.4.1) et l'anglais (section 4.4.2). Les données artificielles sont des exemples explicites extraits automatiquement à partir de données brutes dans lesquelles on supprime le connecteur. Pour l'anglais, on considère également un corpus artificiel constitué à partir des exemples explicites du PDTB. Les données naturelles que nous utilisons sont des exemples implicites provenant des corpus ANNODIS et PDTB décrit dans la section 2.5. Comme les exemples implicites ne sont pas distingués des exemples explicites dans ANNODIS, nous avons dû mettre en place une heuristique pour les identifier. Nous présentons cette heuristique en même temps que les données artificielles. Dans la section suivante, nous présentons les stratégies utilisées dans cette thèse inspirées du cadre de l'adaptation de domaine et les résultats obtenus.

4.4.1 Langue française

Dans cette partie, nous décrivons les données utilisées pour les expériences sur la langue française, et notamment la construction des données artificielles. Nous avons choisi de nous restreindre à quatre relations : *Contrast*, *Result*, *Continuation* et *Explanation*. Ces relations sont annotées dans le corpus français utilisé et correspondent à des exemples implicites, pour lesquels on voudrait

améliorer les performances, et explicites, ce qui permet de constituer un corpus de données artificielles. De plus ce sont quatre des cinq relations (*Summary* n'est pas annotée dans ANNODIS) utilisées dans (SPORLEDER et LASCARIDES, 2008), ce qui nous permet une comparaison mais non directe puisque la langue et le corpus sont différents.

4.4.1.1 Données naturelles

Nous avons déjà décrit les caractéristiques du corpus ANNODIS dans la section 2.5.1.1. Nous disposons pour ce corpus d'un étiquetage en catégorie morpho-syntaxique, d'une lemmatisation et d'indications morphologiques (temps, personne, genre, nombre), pré-traitements effectués avec le MElt tagger (DENIS et SAGOT, 2009). Le MSTParser (MCDONALD et PEREIRA, 2006 ; CANDITO et al., 2010) fournit une analyse en dépendances.

Afin de restreindre notre étude aux exemples implicites, nous nous servons de LexConn, décrit dans la section 2.5.2.2. Nous utilisons une méthode simple : nous projetons le lexique (sauf la forme *à* jugée trop ambiguë) sur les données, ce qui nous permet d'identifier tout token correspondant à un connecteur. Nous ne contraignons pas cette identification sur des critères de position. Cette méthode nous assure d'identifier tous les connecteurs donc de ne récupérer que des exemples implicites mais comporte le risque d'en perdre certains.

Sur les 1 108 exemples disponibles pour les 4 relations nous disposons de 494 exemples implicites ; la distribution des exemples par relation est résumée dans le tableau 4.2. Nous avons fusionné les méta-relations avec les relations correspondantes avec l'hypothèse qu'elles mettaient en jeu le même genre d'indices et de constructions comme ça semble être le cas dans les exemples (52a) et (52b) provenant du LexConn et correspondant respectivement à un exemple de *Result* * et de *Explanation* *.

(52) a. [Pierre a l'air très fatigué,] (*donc/alors*) [il n'a pas dû beaucoup dormir cette nuit.]

b. [Ne t'approche pas du feu,] (*car*) [tu risques de te brûler.]

Relation	Exemples explicites	Exemples implicites	Total
Contrast	100	42	142
Result	52	110	162
Continuation	404	272	676
Explanation	58	70	128
Total	614	494	1 108

Table 4.2.: Corpus ANNODIS : nombre d'exemples explicites et implicites par relation.

4.4.1.2 Données artificielles

Pour la constitution du corpus artificiel, nous avons également utilisé LexConn pour extraire automatiquement des exemples de l'une des 4 relations discursives considérées dans le corpus composé d'articles de l'*Est Républicain* (9M de phrases), avec les mêmes pré-traitements que pour ANNODIS.

Le lexique LexConn contient en tout 329 formes, dont 131 ne peuvent exprimer qu'une seule relation parmi les 4 que nous avons choisies. Les connecteurs ambigus en termes de la relation déclenchée pourraient aussi être intéressants mais ils amèneraient du bruit dans l'annotation automatique puisque nous ne disposons pas de modèle pour les désambiguïser. Nous avons fusionné les relations et les méta-relations correspondantes comme pour les données naturelles.

Nous avons aussi regroupé les 3 relations de type contrastif définies dans ce lexique [6] comme cela a été fait dans le corpus manuellement annoté. Nous n'avons pas pris en compte 3 connecteurs qui ne sont associés à aucune catégorie morpho-syntaxique dans le lexique car leur catégorie morpho-syntaxique est inconnue : *d'où que*, *à en*, *d'où*. Après une première évaluation, nous avons choisi de supprimer 6 connecteurs particulièrement ambigus en emploi pour lesquels il était difficile de définir des motifs départageant emploi discursif et non discursif : *à*, *et*, *maintenant*, *subséquemment*, *dans le coup* et *par comparaison*. Nous disposons finalement de 122 connecteurs dont 100 ont effectivement été utiles finalement, les 22 non utilisés n'ont soit pas été trouvés dans le corpus soit pas dans une configuration définie comme discursive. Par comparaison, SPORLEDER et LASCARIDES (2008) utilisent 55 connecteurs.

L'heuristique d'annotation automatique des exemples se fait en deux étapes. Nous identifions d'abord les formes en emploi discursif en utilisant des motifs (voir tableau 4.3) définis manuellement pour chaque forme de connecteur. Ces motifs sont fondés sur la position de la forme dans la clause hôte, l'argument auquel il est syntaxiquement attaché, parmi initiale, médiane et finale, sa catégorie morpho-syntaxique (POS) et la ponctuation autour de la forme. Nous utilisons également les indications de LexConn qui peuvent parfois préciser la position dans laquelle la forme apparaît en emploi discursif.

Positions	POS	Motifs	Exemples
Inter-phrastique	Toutes POS	A1. **C**(,) A2.	A1. **Malheureusement**(,) A2 A1. **Surtout**, A2.
	Adv.	A1. beg-A2(,) **C**(,) end-A2. A1. A2, **C**.	A1. beg-A2, **de plus**, end-A2. A1. beg-A2(,) **en outre**(,) end-A2. A1. A2, **remarque**.
Intra-phrastique	Toutes POS	A1, **C**(,) A2.	A1, **de plus**(,) A2. A1(,) **donc**(,) A2.
	CS et Prep.	**C** A1, A2.	**Preuve que** A1, A2. **Puisque** A1, A2.
	Adv.	A1, beg-A2(,) **C** (,) end-A2. A1, A2, **C**.	A1, beg-A2, **de plus**, end-A2. A1, beg-A2(,) **en outre**(,) A2. A1, A2, **réflexion faite**.

Table 4.3.: Les motifs définis pour la constitution du corpus artificiel français et quelques exemples correspondants. « A1 » correspond au premier argument, « A2 » au second et « C » au connecteur ; « beg » et « end » correspondent respectivement au début et à la fin d'un argument ; « (x) » indique le caractère facultatif de « x », selon la forme des connecteurs mis en jeu. Certains motifs ne sont applicables que pour certaines catégories morphosyntaxiques de connecteur (« POS ») entre conjonction de subordination (« CS »), préposition (« Prep. ») et adverbe (« Adv. »).

Nous identifions ensuite les arguments d'un connecteur, ce qui peut être vu comme une simplification du problème de segmentation. Nous faisons les mêmes hypothèses simplificatrices que dans les études précédentes : les arguments sont adjacents et couvrent au plus une phrase ou, dans le cas intra-phrastique, on a au plus deux EDU par phrase (MARCU et ECHIHABI, 2002 ; SPORLEDER et LASCARIDES, 2008). Nous utilisons aussi la position du connecteur, sa catégorie morpho-syntaxique et la ponctuation pour cette identification. Lorsque deux connecteurs sont présents dans un segment, il peut arriver que l'un modifie l'autre ou que l'on ait en fait un double connecteur (par exemple « *mais* **parce qu'**il est... »). Dans ce cas, nous risquons de récupérer les mêmes arguments pour deux formes déclenchant des relations différentes ce qui est problématique pour un système de classification. Quand deux connecteurs sont identifiés dans une phrase, nous générons deux exemples à condition que les arguments soient différents : un exemple doit correspondre à une relation s'établissant entre deux phrases et l'autre à une relation s'établissant entre deux propositions à l'intérieur d'une phrase.

6. Les trois relations sont *Contrast*, *Opposition* et *Concession* (ROZE, 2009).

Cette méthode simple permet de générer rapidement de gros volumes de données : au total, nous avons pu extraire 392 260 exemples (voir tableau 4.4, colonne « Disponible »). Nous avons équilibré le corpus en relations en conservant le maximum d'exemples disponibles et divisé les données en un corpus d'entraînement (80 % des données), un de développement (10 %) et un de test (10 %).

Comme nous l'avons déjà dit, on peut observer des différences importantes de distribution entre les données naturelles et artificielles : *Continuation* la plus représentée dans les naturelles devient la moins représentée dans les artificielles.

Notons finalement que cette méthode génère du bruit. SPORLEDER et LASCARIDES (2008) avaient identifié quatre types d'erreurs potentielles lors de la construction d'un corpus artificiel :

1. La paire de segments est étiquetée avec la mauvaise relation ;
2. La paire de segments est étiquetée par une relation alors qu'il n'y en a pas ;
3. L'un des deux arguments est incorrect ;
4. Les frontières des arguments sont fausses.

Nous avons mené une évaluation du corpus artificiel sur 250 exemples choisis aléatoirement. Nous n'avons pas trouvé de cas d'erreur de type 1, montrant que les connecteurs choisis sont effectivement non ambigus. Nous avons trouvé 18 cas d'emploi non discursif comme c'est le cas dans l'exemple (53a) où la forme *surtout* ne signale pas une *Continuation*. Nous avons également trouvé 3 erreurs de type 3 qui correspondent à des cas où nos motifs n'ont pas permis de repérer un connecteur antéposé : l'argument contenant le connecteur est identifié comme apparaissant après le premier argument alors qu'il apparaît en fait avant, comme dans l'exemple (53b). Enfin, nous avons trouvé 37 erreurs de type 4, un chiffre relativement haut mais ce type d'erreur est le moins grave puisqu'il peut rester suffisamment d'information dans les arguments récupérés. Ces erreurs correspondent à des cas où les hypothèses simplificatrices de segmentation ne sont pas vérifiées, comme dans l'exemple (53c) où le premier argument devrait englober les deux premières phrases ou dans l'exemple (53d) où la phrase contient en fait plus de deux EDU, ou à des cas de discours rapporté comme dans l'exemple (53e).

(53) a. [L'hôte dort au pied du lit, discret, mais « plus encombrant qu'une terre-neuve. », et] [*surtout* beaucoup plus maléfique puisqu'il instillera chaque soir, par quelques mots bien choisis, des cauchemars dans le sommeil naguère serein du mari.]

b. [Pelé, Maradona et Ronaldo ont quelques points en commun.] [Hormis le fait que, chacun à son époque , a été considéré comme le meilleur joueur du monde, tous trois ont commencé à « tripoter » la balle dans des ruelles, loin des belles pelouses du Maracana ou de Buenos aires.]

c. A l'initiative de leur association, plusieurs membres de l'amicale des sapeurs pompiers se sont rendus au foyer des aînés et ont offert la galette des rois aux anciens. [Les participants ont dégusté avec plaisir cette délicieuse pâtisserie partagée dans une bonne ambiance.] [Par ailleurs, une formation aux premiers secours sera organisée courant février par les pompiers.]

d. [Par ailleurs, le locataire est tenu, selon l'article 1728 du code civil, d'user de la chose louée en « bon père de famille », c'est-à-dire d'une part de respecter la destination des lieux et] [d'autre part d'entretenir correctement le bien loué.]

e. [Malgré le poids des ans, l'ancien stagiaire du FC Metz, qui vécut du banc de touche l'exploit de son équipe face à Barcelone au Nou Camp en 1983 (« c'est un souvenir extraordinaire.] [Malheureusement, j'étais jeune et je ne me rendais pas bien compte »), a fait des ravages chez les amateurs.]

Relation	Disponible (%)	Entraînement	Développement	Test	Total
Contrast	252 793 (64, 44%)	23 409	2 926	2 926	29 261
Result	50 298 (12, 82%)	23 409	2 926	2 926	29 261
Continuation	29 261 (7, 46%)	23 409	2 926	2 926	29 261
Explanation	59 910 (15, 27%)	23 409	2 926	2 926	29 261
Total	392 262	93 636	11 704	11 704	117 044

Table 4.4.: Corpus artificiel français constitué à partir des données brutes (*Est Républicain*) : nombre d'exemples par relation.

4.4.2 Langue anglaise

Dans cette partie, nous décrivons les données artificielles utilisées pour les expériences sur l'anglais. Ces données sont soit annotées manuellement dans le cadre du PDTB soit extraites automatiquement.

Les premières études utilisant des données artificielles les récupéraient automatiquement à partir de corpus bruts (MARCU et ECHIHABI, 2002 ; SPORLEDER et LASCARIDES, 2008), stratégie reprise ensuite par exemple dans (RUTHERFORD et XUE, 2015). Après la parution du PDTB, certaines études ont proposé d'utiliser directement les données explicites annotées dans ce corpus comme données artificielles comme dans (WANG et al., 2012). LAN et al. (2013) utilisent quant à eux les deux types de données. L'intérêt d'utiliser les données du PDTB réside dans le fait qu'elles sont annotées manuellement, donc elle ne nécessite pas la définition d'heuristiques pour identifier les connecteurs, leurs arguments et la relation qu'ils déclenchent. On a toujours des différences en termes distributionnels mais plus à cause des heuristiques ou du bruit potentiel. Cependant, cette stratégie conduit à des données artificielles moins nombreuses : on double globalement le nombre de données disponibles à l'entraînement tandis qu'avec des données annotées automatiquement on peut multiplier par cent le nombre d'exemples disponibles. Ces deux jeux de données artificielles correspondent donc à un compromis entre qualité et quantité.

4.4.2.1 Corpus artificiel construit à partir du PDTB

Le corpus artificiel construit à partir du PDTB correspond aux données explicites annotées dans ce corpus dans lesquelles on supprime le connecteur. Concernant la relation annotée pour chaque paire de segments, ZHOU et al. (2010) et LAN et al. (2013) utilisent le sens le plus fréquent ce qui rejoint quelque part la stratégie d'utilisation de connecteurs non ambigus lorsque le corpus artificiel est construit à partir de données brutes. Il nous a cependant semblé plus logique de tirer profit du fait que l'on dispose de données annotées manuellement plutôt que de faire comme si ces données étaient extraites automatiquement. Ainsi, si le connecteur *and* est effectivement annoté dans 97, 93% des cas avec la relation *Expansion*, il nous semble que les quelques cas où il apparaît avec une autre relation, par exemple de type contrastif comme dans (54a) ou causal comme dans (54b), peuvent être particulièrement intéressants puisque c'est un connecteur à sémantique faible, parfois non reconnu comme connecteur : ces exemples en particulier doivent contenir une information supplémentaire du type de celle utilisée dans les exemples implicites. Ce raisonnement s'applique également aux autres connecteurs.

(54) a. [We're in a metro area with millions of Bear fans,] and [only a small number can be accomodated.]

Relation	Entraînement	Développement	Test	Total (%)
Temporal	2 633	519	288	3 440 (18, 64%)
Contingency	2 505	564	181	3 250 (17, 61%)
Comparison	4 209	896	366	5 471 (29, 64%)
Expansion	4 770	1 078	450	6 298 (34, 12%)
Total	14 117	3 057	1 285	18 459

Table 4.5.: Corpus artificiel anglais constitué à partir des données manuelles (PDTB) : nombre d'exemples par relation de niveau 1.

b. [Last year, earnings of the combined companies didn't cover debt service] and [Pinkerton's was forced to borrow $20 million of subordinated debt.]

Nous conservons donc tous les exemples explicites en leur attribuant les arguments et les relations annotés. Comme pour les exemples implicites, nous ne prenons en compte que la première annotation. Nous avons 18 459 exemples explicites au total pour le niveau 1 de relation répartis en un ensemble d'entraînement, un ensemble de développement et un ensemble de test en utilisant le même découpage en sections que pour les implicites. Nous donnons le nombre d'exemples par relation au niveau 1 dans la table 4.5. On note que la distribution en relation des données explicites est très différente de celle des données implicites (tableau 3.4), avec cependant une constante : la classe *Expansion* est la plus représentée dans les deux ensembles de données (34, 12% dans les explicites et 53, 58% dans les implicites. Notons cependant que la distribution est plus déséquilibrée dans le cas des implicites où cette classe représente plus de la moitié des données à elles seules. On a beaucoup plus d'exemples explicites des classes *Temporal* (18, 64%) et *Comparison* (29, 64%) que d'implicites (respectivement 5, 14% et 15, 21%) : nous espérons que l'ajout de données pour ces classes sera donc particulièrement bénéfique.

Pour le niveau 2, on dispose de 16 566 exemples, nombre assez proche des 15 798 exemples implicites pour ce niveau, le nombre d'exemples par relation est indiqué dans la table 4.6. Nous utilisons également le découpage en section utilisé pour les implicites pour constituer les ensembles d'entraînement, de développement et de test. Comme on peut le voir, nous avons très peu d'exemples explicites pour la relation *Pragmatic Cause*, ce qui est probablement la raison pour laquelle Lan et al. (2013) ne la prennent pas en compte. Nous la conservons cependant ce qui nous permet de donner des scores pour les 11 relations prises en compte dans les études pour ce niveau même si, clairement, les données explicites ne devraient pas conduire à des améliorations pour cette relation. On peut également observer au niveau 2 des différences importantes en termes de distribution sur les relations. Certaines relations correspondent à très peu d'exemples explicites par rapport aux implicites, ce qui reflète des préférences comme pour la relation *Cause* exprimée de manière bien plus fréquente sans connecteur (10, 97% d'explicites contre 33, 15% d'implicites), ou la relation *Asynchronous* au contraire exprimée plus fréquemment avec un connecteur (12, 21%) que sans (5, 24%), ou peut-être également le fait qu'une relation ne corresponde qu'à peu de connecteurs connus comme *Restatement* (0, 94% d'explicites contre 25, 05% d'implicites).

4.4.2.2 Corpus artificiel construit automatiquement

Nous construisons également un ensemble de données artificielles à partir du corpus *Bllip* [7] constitué d'articles journalistiques du *Los Angeles Times*, du *Washington Post*, du *New York Times* et du *Reuters*. Il contient 310 millions de mots. Il est étiqueté en catégories morpho-syntaxiques et nous disposons

7. https://catalog.ldc.upenn.edu/LDC2008T13

Relation	Entraînement	Développement	Test	Total (%)
Asynchronous	1 539	307	176	2 022 (12, 21%)
Synchronous	1 090	212	111	1 413 (8, 53%)
Cause	1 365	337	116	1 818 (10, 97%)
Pragmatic Cause	5	2	1	8 (0, 048%)
Contrast	2 825	745	274	3 844 (23, 30%)
Concession	1 015	115	71	1 201 (7, 25%)
Conjunction	3 914	911	387	5 212 (31, 46%)
Instantiation	223	57	22	302 (1, 82%)
Restatement	116	28	11	155 (0, 94%)
Alternative	276	48	27	351 (2, 12%)
List	205	33	2	240 (1, 45%)
Total	12 573	2 795	1 198	16 566

Table 4.6.: Corpus artificiel anglais constitué à partir des données manuelles (PDTB) : nombre d'exemples par relation de niveau 2.

d'une analyse syntaxique en constituants acquise automatiquement (meilleure analyse fournie par l'analyseur Charniak-Johnson).

Au lieu d'utiliser une liste de connecteurs prédéfinis en se limitant aux connecteurs peu ambigus et de définir des motifs pour leur identification comme MARCU et ECHIHABI (2002), SPORLEDER et LASCARIDES (2008) et RUTHERFORD et XUE (2015), nous construisons un modèle d'identification des connecteurs et un modèle de désambiguïsation en relation des connecteurs. Nous construisons également un modèle qui identifie les exemples comme inter- ou intra-phrastiques, donc un modèle simplifié de localisation des arguments. Nous n'avons pas utilisé de stratégie visant à gérer le problème de déséquilibre des classes pour ces tâches, suivant les configurations des études existantes (PITLER et NENKOVA, 2009 ; LIN et al., 2014). Pour l'extraction des arguments, la détermination de leur empan, nous conservons les hypothèses simplificatrices précédentes : si l'exemple est intra-phrastique, la phrase contient exactement deux segments discursifs qui sont les arguments du connecteur, s'il est inter-phrastique, la phrase contenant le connecteur est l'argument 2, celle immédiatement précédente est l'argument 1. Pour les exemples intra-phrastiques, nous nous fondons sur la position du connecteur, sa catégorie et la ponctuation pour séparer les deux segments, on impose également la présence d'un verbe dans chaque segment. Pour les exemples inter-phrastiques, si la phrase contenant le connecteur est la première phrase du document, l'exemple est reconverti en exemple intra-phrastique. L'utilisation de modèles plutôt que d'heuristiques nous permet de simplifier le processus d'annotation automatique au sens où cela ne nécessite pas d'étudier chaque connecteur et de lui attribuer un motif. Nous espérons également obtenir des données moins bruitées, les performances pour les exemples explicites étant assez hautes. De plus, cela nous permet de conserver tous les connecteurs possibles, et notamment les connecteurs considérés comme ambigus qui, nous l'espérons, correspondent à des exemples qui ressemblent plus à des exemples implicites au sens où ils doivent mettre en jeu d'autres types d'indices.

Les modèles ont été construits en utilisant les données explicites du PDTB divisées en trois ensembles d'entraînement, de développement et d'évaluation selon la division proposée par LIN et al. (2014) en ne conservant que la première relation annotée avec un modèle par régression logistique. On utilise un jeu de traits simples sans informations syntaxiques correspondant globalement aux traits ajoutés par LIN et al. (2014) et utilisés dans (JOHANNSEN et SØGAARD, 2013) : le connecteur, les combinaisons entre le connecteur et les mots précédents ou suivants, la catégorie morpho-syntaxique du connecteur, celle des mots précédents et suivants, les combinaisons entre catégorie du connecteur et catégories des mots précédents ou suivants. La catégorie morpho-syntaxique d'un connecteur n'est pas toujours facile à obtenir, PITLER et al. (2008) la décrivent comme la catégorie

Classe	P	R	F_1
Négatif	$93,6$	$96,3$	$95,0$
Positif	$91,2$	$85,3$	$88,1$

Table 4.7.: Modèle de désambiguïsation en emploi des connecteurs, scores de précision (« P »), rappel (« R ») et F_1 par classe.

du nœud couvrant l'ensemble des mots qui constituent le connecteur et uniquement ces mots. Mais, par exemple, le connecteur *as soon as* ne correspond pas en général à un seul nœud couvrant comme on peut le voir dans la figure 4.1. Dans ce cas, nous choisissons la catégorie du premier élément composant le connecteur, ici « RB ».

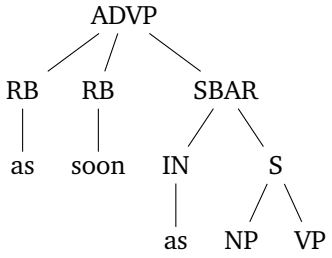


Figure 4.1.: Fragment d'arbre syntaxique pour le connecteur *as soon as* provenant d'un exemple du PDTB.

Pour la désambiguïsation en emploi, donc l'identification des formes en emploi discursif (« Positif ») ou non (« Négatif »), le modèle binaire construit correspond à une exactitude micro-moyennée de $92,9\%$ et à un score de F_1 macro-moyenné de $91,5\%$. Les scores par classe sont donnés dans la table 4.7, la classe dénommée « Positif » correspond aux formes en emploi discursif et la classe « Négatif » aux formes qui ne sont pas en emploi discursif. Le corpus contient 100 formes de connecteurs mais nous ne construisons un modèle de désambiguïsation que pour les connecteurs continus, on laisse de côté les quatre connecteurs discontinus : *either..or* (4 occurrences dans le corpus), *neither..nor* (3 occurrences dans le corpus), *on the one hand..on the other hand* (1 occurrence dans le corpus), et *if..then* (38 occurrences dans le corpus). On a donc finalement 96 connecteurs dans l'ensemble du corpus. L'ensemble d'évaluation contient 923 formes positives et $2\,075$ négatives. Ces scores sont légèrement inférieurs à ceux obtenus dans (Pitler et Nenkova, 2009) ou dans (Lin et al., 2014), ces auteurs rapportant respectivement au mieux une exactitude de $96,26\%$ et de $97,34\%$. Ceci est probablement dû au fait que nous n'utilisons pas d'informations syntaxiques mais cela nous permet de ne pas dépendre de possibles erreurs dans l'analyse syntaxique en construisant les données à partir du *Bllip*.

Lorsque nous passons à la phase d'identification des connecteurs dans le *Bllip*, nous nous ramenons à identifier les 100 connecteurs, en incluant les discontinus. La seconde partie de ces connecteurs discontinus correspond à un connecteur dans le PDTB (*or*, *nor*, *then* and *on the other hand*). Nous pouvons donc identifier ces connecteurs discontinus dans les données brutes en appliquant le modèle de désambiguïsation et en cherchant ensuite l'éventuelle présence de la première partie (les deux parties devant obligatoirement être dans la même phrase, sauf pour *on the one hand..on the other hand*). Le nombre d'exemples extraits pour chaque connecteur est indiqué dans l'annexe A, notons que 4 connecteurs n'ont conduit à l'extraction d'aucun exemple, nous disposons donc d'exemples pour 96 connecteurs.

Le corpus d'évaluation du PDTB contient 546 instances de relations inter-phrastiques et 377 intra-phrastiques. Le modèle de localisation des arguments est également un modèle binaire, nous obtenons pour ce modèle une exactitude de $96,1\%$, seuls 36 exemples sont mal prédits. Lin et al. (2014) rapportent uniquement des scores de F_1 par classe de $98,3\%$ pour la classe intra-phrastique

Classe	P	R	F_1
Intra-phrastique	97,4	96,0	96,7
Inter-phrastique	94,3	96,3	95,3

Table 4.8.: Modèle de localisation des arguments des connecteurs : scores de précision (« P »), rappel (« R ») et F_1 par classe.

Classe	P	R	F_1
Temporal	84,9	95,3	89,8
Contingency	98,7	85,7	91,7
Comparison	97,0	96,7	96,8
Expansion	98,2	99,3	98,7
Micro-moyenne	95,1	-	-
Macro-moyenne	94,7	94,2	94,3

Table 4.9.: Modèle d'identification des relations explicites au niveau 1, scores de précision (« P »), rappel (« R ») et F_1 par relation, nous indiquons également les scores moyennés du système.

et 97,5% pour la classe inter-phrastique. Les scores détaillés pour cette étape sont fournis dans la table 4.8.

Nous effectuons une désambiguïsation en relation aux niveaux 1 et 2 en construisant des modèles multiclasses. Pour le niveau 1, nous obtenons une exactitude de 95,1% supérieure à l'exactitude de 94,1% rapportée dans (PITLER et al., 2008) cependant avec un autre ensemble d'évaluation. Les scores détaillés sont repris dans la table 4.9.

Enfin, pour le niveau 2 de relation, nous obtenons un score d'exactitude de 86,2% soit très proche du score de précision de 86,8% rapporté par LIN et al. (2014). Les scores détaillés sont rapportés dans la table 4.10. Comme on pouvait s'y attendre, la relation *Pragmatic Cause* correspond à un très mauvais score dû au fait qu'elle correspond à très peu d'exemples. On retrouvera donc ce problème dans les données artificielles dans lesquelles nous ne récupérons pas d'exemple de cette relation. On note également que les scores pour la relation *List* et *Concession* sont très bas. La littérature existante à tendance à considérer l'identification des relations explicites comme un problème résolu, étant donné que les scores globaux sont assez hauts. On voit bien cependant ici que ce n'est pas vraiment le cas et que des efforts restent à faire pour certaines relations peu représentées ou correspondant à des distinctions fines.

Relation	P	R	F_1
Asynchronous	94,3	83,0	88,3
Synchronous	63,0	88,7	73,7
Cause	98,0	88,3	92,9
Pragmatic Cause	0,0	0,0	0,0
Contrast	93,1	90,0	91,6
Concession	34,5	43,5	38,5
Conjunction	85,6	97,7	91,2
Instantiation	100,0	100,0	100,0
Restatement	100,0	14,3	25,0
Alternative	88,9	100,0	94,1
List	50,0	3,4	6,5
Micro-moyenne	86,2	-	-
Macro-moyenne	73,4	64,4	63,8

Table 4.10.: Modèle d'identification des relations explicites au niveau 2, scores de précision (« P »), rappel (« R ») et F_1 par relation, nous indiquons également les scores moyennés du système.

Relation	Disponible (%)	Entraînement	Développement	Test
Temporal	783 080 (26, 43%)	252 212	31 493	31 638
Contingency	315 343 (10, 64%)	252 286	31 650	31 407
Comparison	1 148 245 (38, 76%)	252 172	31 575	31 596
Expansion	715 878 (24, 16%)	252 428	31 419	31 496
Total	2 962 546	1 009 098	126 137	126 137

Table 4.11.: Corpus artificiel anglais constitué à partir des données brutes (*Bllip*) : nombre d'exemples par relation de niveau 1.

Relation	Disponible (%)	Entraînement	Développement	Test
Asynchronous	353 616 (11, 94%)	252 556	31 460	31 327
Synchronous	429 464 (14, 50%)	252 245	31 553	31 545
Cause	315 341 (10, 64%)	252 106	31 650	31 587
Pragmatic Cause	0 (0%)	0	0	0
Contrast	781 801 (26, 39%)	252 314	31 453	31 576
Concession	366 445 (12, 37%)	252 254	31 446	31 643
Conjunction	665 812 (22, 47%)	252 234	31 610	31 499
Instantiation	9 (0, 0003%)	7	1	1
Restatement	1 025 (0, 035%)	815	104	106
Alternative	48 818 (1, 65%)	38 999	4 911	4 908
List	215 (0, 0072%)	170	25	20
Total	2 962 546	1 553 700	194 213	194 212

Table 4.12.: Corpus artificiel anglais constitué à partir des données brutes (*Bllip*) : nombre d'exemples par relation de niveau 2.

Nous avons finalement extrait au total 2 962 546 exemples artificiels. Nous donnons dans la table 4.11 le nombre d'exemples artificiels extraits à partir du *Bllip* pour le niveau 1 de relation et dans la table 4.12 pour le niveau 2 de relation. On conserve 315 343 exemples par relation pour constituer le corpus final. On a divisé les données en trois ensembles en respectant la proportion 80% de données d'entraînement et 10% de données de développement et d'évaluation. Pour le niveau 2, nous n'avons aucun exemple de la relation *Pragmatic Cause* qui est donc ignorée dans les expériences avec ce corpus artificiel.

4.5 Adaptation de domaine pour l'identification des relations implicites

Dans cette partie, nous décrivons les stratégies mises en place pour gérer le problème de dissimilarité entre les données naturelles et artificielles (section 4.5.1), stratégies inspirées du cadre de l'adaptation de domaine décrit précédemment et fondées sur une combinaison des données ou des modèles. Nous ne nous plaçons pas exactement dans le cadre de l'adaptation de domaine au sens où nous ne sommes pas intéressée par les performances du modèle construit sur ce que nous considérons comme les données sources, les données artificielles. Seules les performances sur les données cibles, les données naturelles, sont considérées comme importantes même si de bons résultats sur les sources sont un bon indice que les informations utilisées sont aussi pertinentes pour cet ensemble de données, donc que la relation reste identifiable. Nous présentons ensuite les résultats des expériences menées sur le français (section 4.5.2) et sur l'anglais (section 4.5.4).

4.5.1 Stratégies mises en place

Nous avons évalué les méthodes proposées par Daumé III (2007) comme systèmes de référence, des méthodes décrites comme relativement compétitives et simples à mettre en œuvre. Les diffé-

rentes méthodes de combinaison que nous proposons diffèrent selon que la combinaison s'opère directement au niveau des jeux de données ou au niveau des modèles entraînés sur ceux-ci. Nous avons ajouté un principe de sélection automatique des exemples automatiquement annotés basé sur la confiance en sa prédiction du modèle entraîné sur ces données afin de gérer une partie du bruit, sélectionner les exemples les plus sûrs. Les performances de tous les systèmes seront comparées à celles des systèmes de référence entraînés séparément sur les deux ensembles de données.

Systèmes de référence

Le premier système de référence (NATONLY) est celui où l'on s'entraîne et l'on s'évalue sur les données manuellement annotées. Ce système nous donne les performances à dépasser et correspond à un cadre classique d'apprentissage où toutes les données sont supposées tirées identiquement de la même distribution sous-jacente inconnue.

Le second système de référence (ARTONLY) est celui où l'on s'entraîne sur les données artificielles et l'on s'évalue sur les données naturelles. C'est le système proposé dans les études de BLAIR-GOLDENSOHN et al. (2007) et SPORLEDER et LASCARIDES (2008) et on devrait normalement observer comme dans ces études une baisse importante des performances.

Ces systèmes servent de comparaison aux autres systèmes testés, des systèmes où sont combinés les données ou les modèles.

Combinaison des données

La première stratégie de combinaison que nous étudions (UNION) relève du premier type : elle consiste à créer un corpus d'entraînement qui contient la réunion des deux ensembles de données. Cette méthode ne permet pas de gérer l'importance de chacun des deux ensembles de données ou d'en estimer l'impact sur le système. Nous la déclinons donc en deux autres méthodes.

Une stratégie dérivée (ARTSUB) consiste à prendre, non pas l'intégralité des données artificielles, mais des sous-ensembles aléatoires de ces données, en addition des données manuelles. Cette méthode est un peu plus subtile dans la mesure où l'on peut faire varier la proportion des exemples artificiels par rapport aux exemples manuels. On ne prend cependant pas en compte l'intégralité des données artificielles donc on perd de l'information.

Enfin, la troisième méthode du premier type (NATW) garde cette fois la totalité des données artificielles mais pondère (ou duplique) les exemples manuels de manière à éviter un déséquilibre trop grand au profit des données artificielles.

Combinaison des modèles

Dans le second type de méthodes, on trouve tout d'abord une méthode (ADDPRED) qui consiste à utiliser les prédictions d'un modèle entraîné sur les données artificielles (à savoir les données « sources ») comme descripteur dans le modèle entraîné sur les données naturelles (à savoir les données « cibles »). Le paramètre associé à ce descripteur mesure donc l'importance à accorder aux prédictions du modèle entraîné sur les données artificielles. Cette méthode est le meilleur système de référence et le troisième meilleur modèle dans (DAUMÉ III et MARCU, 2006).

Une variation de cette méthode (ADDPROB) utilise en plus le score de confiance (par exemple, la probabilité) du modèle artificiel comme descripteur supplémentaire dans le modèle manuel. L'idée

à la base de ces méthodes est que même si le classifieur se trompe, il est peut-être consistant dans ses erreurs qui peuvent se révéler une source d'information utile pour le modèle.

Une troisième méthode (ARTINIT) vise à initialiser les paramètres du modèle entraîné sur les données manuelles avec ceux du modèle utilisant les données artificielles.

Enfin, la dernière méthode (LININT) se base sur une interpolation linéaire de deux modèles préalablement entraînés sur chacun des ensembles de données. Elle correspond donc à une combinaison des paramètres des modèles avec une pondération de chacun des modèles à l'aide du coefficient intervenant dans l'interpolation linéaire.

Sélection automatique d'exemples

Nous avons aussi testé toutes les stratégies en ajoutant une étape de sélection automatique d'exemples artificiels. La méthode utilisée se fonde simplement sur la probabilité du label prédit : on teste différents seuils sur ces probabilités en ajoutant à chaque fois les seuls exemples prédits avec une probabilité supérieure au seuil, et en rééquilibrant éventuellement l'ensemble des données. Cette sélection vise à écarter des données bruitées, en explorant finalement l'une des voies proposées par MARCU et ECHIHABI (2002) et développée d'une autre manière par BLAIR-GOLDENSOHN et al. (2007), à savoir améliorer la qualité du corpus artificiel. Cette méthode permet aussi dans une certaine mesure de sélectionner les données pour lesquelles l'hypothèse de redondance du connecteur est vérifiée : plus le modèle est confiant dans sa prédiction, plus on a de chance qu'il ait trouvé de bons indices pour sa prédiction.

4.5.2 Expériences sur le français

Nous présentons dans cette section les expériences menées sur le corpus français ANNODIS. Ces expériences correspondent à l'évaluation des stratégies inspirées de l'adaptation de domaine présentées dans la section précédente en utilisant le corpus artificiel annoté automatiquement à partir du corpus *Est Républicain*. Les sections suivantes seront consacrées aux expériences sur le corpus anglais du PDTB à partir des données artificielles manuellement et automatiquement annotées.

4.5.2.1 Représentation des données

Notre jeu de traits se fonde sur les travaux existants avec quelques adaptations notables pour le français. Ces traits exploitent des informations de surface, ainsi que d'autres issues d'un traitement linguistique plus profond. Par comparaison, MARCU et ECHIHABI (2002) ne se fondent que sur la co-occurrence de mots dans les segments. SPORLEDER et LASCARIDES (2005b) montrent que la prise en compte de différents types de traits linguistiquement motivés améliore les performances. SPORLEDER et LASCARIDES (2008) utilisent des traits variés dont des bi-grammes de lemmes mais sans traits syntaxiques. Nous avons testé des traits lexico-syntaxiques utilisés dans les précédentes études sur cette tâche. Nous n'avons pas pu reprendre les traits sémantiques comme les classes sémantiques des têtes des arguments car les ressources nécessaires n'existent pas pour le français.

Certains traits sont calculés pour chaque argument :

1. Indice de complexité syntaxique : nombre de syntagmes nominaux, verbaux, prépositionnels, adjectivaux, adverbiaux (valeur continue)
2. Information sur la tête d'un argument :
   — Lemme d'éléments négatifs sur la tête (booléen)

— Information temporelle/aspectuelle : nombre de fois où un lemme de fonction auxiliaire dépendant de la tête apparaît (valeur continue), temps, personne, nombre de l'auxiliaire (booléen)

— Informations sur les dépendants de la tête : présence d'un objet, par-objet, modifieur ou dépendant prépositionnel de la tête, du sujet ou de l'objet (booléen) ; catégorie morpho-syntaxique des modifieurs et des dépendants prépositionnels de la tête, du sujet ou de l'objet (booléen)

— Informations morphologiques : temps et personne de la tête verbale, genre de la tête non verbale, nombre de la tête, catégorie morpho-syntaxique précise (par exemple "VPP") et simplifiée (respectivement "V") (booléen)

D'autres traits portent sur la paire d'arguments :

1. Trait de position : si l'exemple est inter ou intra-phrastique (booléen)
2. Indice de continuité thématique : chevauchement en lemmes et en lemmes de catégorie ouverte (continue)
3. Information sur les têtes des arguments :
   — Paire des temps des têtes verbales (booléen)
   — Paire des nombres des têtes (booléen)

On notera finalement que notre but portant avant tout sur la combinaison de données, nous n'avons pas cherché à optimiser ce jeu de traits, ce qui aurait introduit un paramètre supplémentaire dans notre modèle.

4.5.2.2 Algorithme et évaluation

Les expériences sont réalisées avec l'implémentation de l'algorithme par maximum d'entropie fournie dans la librairie MegaM en version multiclasse avec au maximum 100 itérations. Nous n'avons pas cherché à optimiser les hyper-paramètres correspondant à la régularisation : la norme utilisée est la norme L_2, le coefficient de régularisation vaut 1 par défaut (voir formule 3.3, le coefficient de régularisation correspond ici à la précision de la prior gaussienne : $\lambda = \frac{1}{\sigma^2}$).

Nous utilisons un corpus de données naturelles équilibré à 70 exemples au maximum par relation. Il faudra envisager des expériences conservant la distribution naturelle des données, très déséquilibrée, mais pour l'instant nous nous focalisons sur l'aspect combinaison des données. Nous rappelons le nombre de données disponibles dans la table 4.13.

Relation	Données naturelles	Données artificielles		
		Entraînement	Développement	Test
Contrast	42	23 409	2 926	2 926
Result	70	23 409	2 926	2 926
Continuation	70	23 409	2 926	2 926
Explanation	70	23 409	2 926	2 926
Total	252	93 636	11 704	11 704

Table 4.13.: Expériences sur le français : nombre d'exemples dans les données naturelles, provenant du corpus français ANNODIS et dans les données artificielles, construites à partir de l'*Est Républicain*, utilisées pour la construction et l'évaluation des modèles.

Étant donné le faible volume de données naturelles manuellement annotées disponibles, il n'est pas possible de découper celles-ci en ensembles distincts pour l'entraînement et la validation. En conséquence, nous utilisons une validation croisée dite « enchâssée » (*nested cross-validation*). Cette méthodologie permet de s'assurer que l'optimisation des hyper-paramètres (décrits en section 4.5.2.3) se fait sur des données différentes de celles sur lesquelles on s'évalue. Il a été montré

que cette méthode permet d'obtenir une estimation réaliste de l'erreur (SCHEFFER, 1999 ; VARMA et SIMON, 2006). Comme son nom l'indique, la validation croisée enchâssée comporte deux boucles : une boucle interne de validation croisée qui sert à optimiser le ou les hyper-paramètres, donc à sélectionner le modèle, et une boucle externe de validation croisée qui sert à mesurer les performances obtenues, donc à estimer son erreur.

La boucle externe parcourt les N sous-ensembles du premier découpage aléatoire. Le sous-ensemble k, avec $1 \leq k \leq N$, est considéré comme l'ensemble d'évaluation et les $N-1$ autres sous-ensembles servent à la sélection du modèle dans la boucle interne : l'ensemble des données correspondant à ces $N-1$ sous-ensembles est découpé aléatoirement en M sous-ensembles et une validation croisée sur ces M sous-ensembles permet de déterminer les meilleures valeurs pour les hyper-paramètres. Un classifieur est enfin entraîné avec les $N-1$ sous-ensembles de la boucle externe et évalué sur le sous-ensemble k en utilisant les meilleures valeurs pour les hyper-paramètres. Nous avons utilisé ici 2 validations croisées en 5 sous-ensembles pour déterminer et évaluer les meilleurs modèles pour chacun des systèmes décrits précédemment. Il n'est donc pas garanti que les meilleurs soient sélectionnés à chaque étape de test, mais cette méthode permet d'évaluer la stabilité du système par rapport au choix des hyper-paramètres (*i.e.*, les valeurs ne doivent pas être trop éparpillées), le sur-entraînement (*i.e.*, si les estimations dans les boucles internes et externes sont très éloignées, c'est que la méthode d'optimisation conduit à un sur-entraînement) et la stabilité des modèles (*i.e.*, on s'intéresse à la variance dans la capacité prédictive, sur les résultats sur les sous-ensembles de la boucle externe). Les systèmes n'utilisant aucun hyper-paramètre sont simplement évalués dans la boucle externe.

Comme dans les études précédentes, les performances sont données en termes d'exactitude globale (micro-moyenne) sur l'ensemble des relations, des scores ventilés de mesure F_1 par relation sont également fournis. Ici, les données sont quasiment équilibrées en relation, l'exactitude est donc une mesure pertinente. Dans le cas de très petits échantillons (ici $N = 5$), il est difficile de calculer la significativité des écarts de performance observés. WINTER (2013) montre que le test de Student apparié peut être utilisé dans ce cas, à condition que l'effet de taille (*effect size*, calculé avec le coefficient de Cohen) et que la corrélation entre les échantillons soient suffisamment importants. WINTER (2013) indique qu'au contraire, le test des rangs signés de Wilcoxon peut conduire à des p-valeurs trop hautes avec de si petits échantillons. Nous choisissons donc de donner les résultats en utilisant le test de Student (avec une p-valeur $< 0{,}05$).

4.5.2.3 Résultats sur le français

Modèles de référence

Dans un premier temps, nous construisons deux modèles distincts, l'un à partir des seules données naturelles (NATONLY, 252 exemples), l'autre à partir des seules données artificielles (ARTONLY, $93\,636$ exemples d'entraînement). Notre modèle NATONLY obtient une exactitude de $37{,}3$ %, avec des scores de F_1 par relation compris entre $15{,}0$ % pour *Contrast* et $47{,}9$ % pour *Explanation* (voir tableau 4.14). La relation *Contrast* est donc très mal identifiée peut-être parce que sous-représentée, seulement 42 exemples contre 70 pour les autres relations, le manque de données joue probablement ici un rôle important.

Le modèle ARTONLY obtient une exactitude de $47{,}8$ % lorsqu'évalué sur le même type de données ($11\,704$ exemples de test), mais de $23{,}0$ % lorsqu'évalué sur les données naturelles (voir tableau 4.14). Cette baisse importante est comparable à celle observée dans les études précédentes sur l'anglais. Elle s'explique par les différences de distribution étudiées en section 4.3.3. De manière générale, nous observons des dégradations par rapport à NATONLY pour l'identification de *Result* et

	NatOnly	ArtOnly	
Données de test	Naturelles	Naturelles	Artificielles
Exactitude	37,3	23,0	47,8
Contrast	15,0	23,2	38,3
Result	47,6	15,7	57,4
Continuation	28,1	32,1	54,3
Explanation	47,9	22,4	37,5

Table 4.14.: Modèles de référence sur le français : exactitude du système et F_1 par relation

	Union	NatW	ArtSub	AddPred	AddProb	ArtInit	LinInt
Exactitude	22, 6	38, 9	34, 5	39, 3	38, 9	40, 1	39.3
Contrast	22,0	20,3	15,0	16,0	15,6	16,9	17,1
Result	15,2	40,4	39,4	50,6	48,0	45,9	45,4
Continuation	38,2	44,7	36,2	31,9	31,9	34,0	38,0
Explanation	15,7	42,2	39,2	46,7	48,9	52,2	47,5

Table 4.15.: Modèles avec combinaison sur le français, exactitude du système et F_1 par relation

Explanation (voir tableau 4.14). En revanche l'identification de *Contrast* présente une amélioration, obtenant 23,2 % de F_1 avec 11 exemples correctement identifiés contre 6 précédemment.

Modèles avec combinaisons de données

Dans cette section, nous présentons les résultats des systèmes qui exploitent à la fois les données naturelles et les données artificielles. Ces ensembles de données sont ou bien combinés directement ou bien donnent lieu à des modèles séparés qui sont combinés plus tard.

Certains de ces modèles utilisent des hyper-paramètres. Ainsi, pour la pondération des exemples naturels nous testons différents coefficients de pondération c avec $c \in [0{,}5, 1, 5]$ et $c \in [10\,;2\,000]$ avec un incrément de 10 jusqu'à 100, de 50 jusqu'à 1 000 et de 500 jusqu'à 2 000. Pour l'ajout de sous-ensembles des données artificielles, nous ajoutons à chaque fois n exemples parmi ces données où $n = k$ fois le nombre de données naturelles disponibles avec $k \in [0{,}1\,;600]$ avec un incrément de 0,1 jusqu'à 1, de 10 jusqu'à 100 et de 50 jusqu'à 600. Enfin, pour l'interpolation linéaire des modèles, nous construisons un nouveau modèle en pondérant le modèle artificiel avec $\alpha \in [0{,}1\,;0{,}9]$ avec des incréments de 0,1 (dans le nouveau modèle, nous pondérons donc le modèle construit sur les données naturelles par un coefficient de $1 - \alpha$). Les scores des systèmes sont repris dans le tableau 4.15.

De manière générale, l'ensemble de ces systèmes avec les bons hyper-paramètres conduit à des résultats au moins équivalents et parfois supérieurs en exactitude par rapport à NatOnly. Si la tendance générale est plutôt d'une hausse des performances, aucune des différences observées à ce stade ne semble cependant être statistiquement significative. Le meilleur score d'exactitude est obtenu avec le système ArtInit (40,1 % d'exactitude, p-valeur de 0,18 avec un effet de taille faible, 0,39). Deux autres systèmes obtiennent un score d'exactitude supérieur à 39 %, AddPred et LinInt, avec des résultats toujours non significatifs. Le modèle similaire à AddPred, mais exploitant en plus les probabilités (AddProb), mène à une légère diminution ce qui suggère que les traits de probabilité dégradent les performances. Pour ces systèmes, les scores d'exactitude obtenus sur chaque sous-ensemble sont très proches [8], notamment pour AddPred, ce qui suggère une bonne stabilité des modèles. Les autres systèmes nous permettent d'évaluer l'effet des données artificielles dans les résultats finaux.

8. Écart-type de 0,074 pour ArtInit pour une moyenne de 40,1 et de 0,037 pour AddPred et 0,061 pour AddProb pour une moyenne d'environ 39.

La seule configuration qui mène à des résultats négatifs est l'union simple des corpus d'entraînement (UNION). Ce système obtient 22,6 % d'exactitude donc de l'ordre d'un entraînement sur les seules données artificielles. Ces résultats ne sont pas surprenants : les données naturelles étant environ 372 fois moins nombreuses que les données artificielles, elles se retrouvent pour ainsi dire « noyées » dans les données artificielles.

Les expériences de combinaison des données, ajout de sous-ensembles aléatoires des données artificielles (ARTSUB, 34,5 % d'exactitude) et pondération des exemples naturels (NATW, 38,9 % d'exactitude), montrent l'influence de l'importance relative entre les deux types de données sur l'exactitude des systèmes. Nous pouvons observer cet effet en regardant les scores obtenus dans la boucle interne d'optimisation : pour ces deux systèmes, les données naturelles doivent avoir un poids environ 2,5 fois supérieur aux données artificielles pour obtenir les meilleurs scores d'exactitude. La variance entre les valeurs d'hyper-paramètres choisies est très haute pour ARTSUB, ceci est probablement dû au caractère aléatoire du choix des sous-ensembles qui peut conduire à des différences importantes. Elle est un peu moins forte pour NATW, montrant que cette méthode est plus robuste, mais le choix *a priori* d'une valeur pour l'hyper-paramètre reste large (1 020 à plus ou moins 272). L'interpolation linéaire (LININT) des deux modèles permet d'observer le même effet. Les meilleurs modèles sont obtenus avec un coefficient α en moyenne égal à 0,3, donc de nouveau une influence plus de deux fois plus forte. L'écart-type est relativement élevé mais il semblerait raisonnable de choisir *a priori* pour un futur modèle une valeur correspondant à la moyenne.

Les méthodes de combinaison aboutissent à des systèmes d'exactitude similaire voire supérieure à NATONLY. Au niveau des scores par relation, nous avons observé lors de la phase d'optimisation, qu'une influence forte des données artificielles avait tendance à améliorer l'identification de *Contrast* et de *Continuation*. Ainsi la moyenne de la F_1 augmente avec l'augmentation du coefficient α et l'interpolation linéaire des modèles. Au contraire, un poids fort des données artificielles entraîne une forte dégradation de l'identification de *Result* et *Explanation*. La relation *Contrast* profite peut-être de données artificielles moins bruitées : la majorité des exemples (plus de 75 %) sont construits à partir de *mais*, une forme qui est toujours en emploi discursif et dont les arguments sont dans l'ordre canonique, « argument1 + connecteur + argument2 ». Les différences de performance au niveau des classes peuvent venir de distributions plus ou moins proches entre les deux types de données. En regardant la distribution en terme de traits (850 traits en tout), nous constatons un écart de plus de 30 % pour 2 et 5 traits pour *Result* et *Explanation* respectivement mais aucun pour *Contrast* et *Continuation*, les relations pour lesquelles l'apport direct des données artificielles est positif.

Modèles avec sélection automatique d'exemples

Les expériences précédentes ont montré que l'ajout de données artificielles donnait le plus souvent lieu à des gains de performance, mais ces gains restent relativement modestes, voire non significatifs. Notre hypothèse est que de nombreux exemples artificiels amènent du bruit dans le modèle. Idéalement, nous souhaiterions être capable de sélectionner les exemples artificiels les plus informatifs et qui complémentent le mieux les données naturelles.

La méthode de sélection d'exemples que nous proposons a pour objectif d'éliminer les exemples potentiellement plus bruités. Pour cela, le modèle artificiel est utilisé sur les données d'entraînement et nous conservons les exemples prédits avec une probabilité supérieure à un seuil $s \in [0{,}3\ ;0{,}85]$ avec un incrément de 0,1 jusqu'à 0,5 et de 0,05 jusqu'à 0,85. Si ce modèle est assez sûr de sa prédiction, on peut espérer que l'exemple ne correspond pas à du bruit, à une forme en emploi non discursif et/ou à une erreur de segmentation. Nous vérifions aussi, en quelque sorte, l'hypothèse de redondance du connecteur. Pour chaque seuil, nous rééquilibrons les données en nous fondant sur

+ SELEC	UNION	NATW	ARTSUB	ADDPRED	ADDPROB	ARTINIT	LININT
Exactitude	40,1	41,3	39,3	**41,7***	35,7	36,9	36,5
Contrast	**25,9**	19,2	21,6	20,8	14,5	18,9	17,5
Result	45,3	48,3	44,4	**51,0**	41,1	34,5	38,2
Continuation	34,8	32,4	33,8	31,2	30,2	31,0	**37,8**
Explanation	48,9	53,4	47,6	**53,9**	45,3	52,8	44,3

Table 4.16.: Modèles avec sélection d'exemples sur le français, exactitude du système et F_1 par relation ; * signale un résultat significativement supérieur à NATONLY

la relation la moins représentée (système + SELEC). Les scores des systèmes sont repris dans le tableau 4.16.

La sélection automatique d'exemples permet d'améliorer la plupart des résultats précédents, montrant l'intérêt de cette étape supplémentaire. En particulier, le système correspondant à l'ajout de traits de prédiction (ADDPRED + SELEC) correspond à une claire tendance vers une amélioration significative en termes d'exactitude par rapport à un entraînement sur les seules données naturelles (p-valeur de $0{,}033$ avec un effet de taille important de $0{,}756$ et une corrélation forte, $0{,}842$). Les scores de F_1 pour toutes les classes sont améliorés : $20{,}8$ % pour *Contrast*, $51{,}0$ % pour *Result*, $31{,}2$ % pour *Continuation* et $53{,}9$ % pour *Explanation*. Deux autres systèmes obtiennent un score d'exactitude supérieur à 40 % : NATW + SELEC ($41{,}3$ %, avec une tendance vers une amélioration significative[9]) et UNION + SELEC ($40{,}1$ %, non significativement meilleur). Notons que le système ADDPRED correspond au meilleur système de référence dans (DAUMÉ III et MARCU, 2006), ce qui tend à montrer que prendre en compte la différence de distribution entre nos données sous l'angle de l'adaptation de domaine est pertinent.

De manière générale, la phase de sélection permet d'autoriser un poids plus fort sur les informations provenant des données artificielles. Pour le système LININT + SELEC, les meilleurs résultats sont obtenus avec une influence quasiment égale des deux modèles. De même, pour NATW + SELEC, la moyenne des coefficients choisis est beaucoup plus basse, et elle augmente largement pour ARTSUB + SELEC, autorisant des sous-ensembles plus larges, avec cependant un fort éparpillement des valeurs. Ces considérations accréditent malgré tout l'hypothèse selon laquelle la sélection améliore la qualité du corpus artificiel. Au niveau des seuils choisis, la moyenne sur toutes les expériences est aux alentours de $0{,}7$, avec un écart-type variable selon les systèmes mais supérieur à $0{,}1$. C'est un écart assez important, cet hyper-paramètre semble relativement mal estimé par notre validation croisée interne et nécessiterait de futures expériences, où l'on pourra réduire l'espace de recherche.

La sélection des exemples améliore l'identification des relations et conduit à un système améliorant significativement l'exactitude de NATONLY montrant que les données artificielles lorsque intégrées de façon adéquate peuvent améliorer l'identification des relations implicites, notamment lorsque leur influence est faible, le modèle étant guidé vers la bonne distribution.

À la constitution des corpus avec sélection, nous avons observé qu'avec la croissance du seuil, nous conservions toujours plus d'exemples pour *Result*. Dès le seuil $0{,}4$, nous disposons d'environ $3\ 900$ exemples de plus pour cette relation, alors que *Contrast* devient sous-représentée. Cette observation montre que le bruit n'est probablement pas la seule façon d'expliquer les résultats puisque la relation améliorée par les données artificielles est celle pour laquelle le modèle artificiel est le moins confiant alors que la relation dont les résultats sont les plus dégradés est celle pour laquelle le modèle est le plus confiant.

9. p-valeur de $0,077$, effet de taille large de $0{,}68$ et importante corrélation $0{,}67$.

Le corpus français est de petite taille ce qui nous a conduit à utiliser une méthode d'évaluation particulière, la validation croisée enchâssée. Si cette méthode nous permet de conclure qu'il est possible d'améliorer les performances globales, elle rend difficile une application pour de nouvelles données puisque chaque modèle optimal choisi au sein de la boucle interne peut utiliser des paramètres différents. De plus, si nous avons pu observer des améliorations, nous aimerions savoir si c'est également le cas sur les données du PDTB plus largement utilisées. Nous menons donc des expériences similaires à partir des données de ce corpus anglais.

4.5.3 Expériences sur l'anglais, corpus artificiel PDTB

Nous avons vu que les méthodes de combinaison inspirées de l'adaptation de domaine permettaient d'obtenir sur le français des améliorations significatives pour un système multiclasse construit pour quatre relations. Dans cette partie, nous évaluons ces méthodes sur le corpus du PDTB en utilisant le corpus artificiel manuel obtenu également à partir du PDTB. Dans la partie suivante, nous présenterons les résultats obtenus avec un corpus artificiel annoté automatiquement.

4.5.3.1 Configuration des expériences

Représentation des données

Nous testons deux représentations : celle fondée sur les seules paires de mots et celle correspondant à l'ensemble **base+lex** présentée dans le chapitre précédent qui regroupe des informations sémantiques et lexicales. On n'utilise pas ici les règles de production syntaxiques. Nous avons vu dans les résultats de référence dans le chapitre précédent que l'ensemble **base+lex** permettait des résultats similaires voire supérieurs à ceux obtenus en ajoutant les traits syntaxiques. De plus, nous pensons que ces traits syntaxiques sont particulièrement inadaptés dans le cas d'utilisation de données artificielles, puisque nous supprimons une expression dans les instances ce qui modifie l'analyse syntaxique. Il faudrait probablement également chercher des solutions pour réduire l'éparpillement pour ce type de traits par exemple en utilisant un encodage différent de cette information. Cependant, ici nous nous concentrons sur l'éparpillement des paires de mots. Notons que dans la configuration multiclasse de niveau 1, un modèle entraîné sur les données artificielles avec les traits de type règles syntaxiques conduit à des performances de l'ordre de $33,17\%$ en exactitude et $25,03\%$ en macro-F_1 donc inférieures à celles obtenues avec les paires de mots qui correspondent à $35,37\%$ d'exactitude et $28,24\%$ de macro-F_1. Ce jeu de traits engendre notamment une baisse de près de 7% pour la relation *Contingency*, et de 2 à 3% pour les autres relations ce qui, nous semble-t-il, confirme l'inadéquation de cette représentation pour notre configuration. Rappelons que MARCU et ECHIHABI (2002) utilisaient uniquement les paires de mots tandis que les études suivantes sur cette tâche ont montré l'intérêt de l'utilisation de traits plus linguistiquement motivés.

Modèle

On utilise un algorithme de régression logistique implémenté dans Scikit-Learn (PEDREGOSA et al., 2011) comme dans les expériences de référence présentées dans le chapitre précédent. On optimise les mêmes hyper-paramètres, type de norme, entre L_1 et L_2, et force de la régularisation (voir section 3.4.3). On fixe par contre ici un filtre en fréquence sur les traits, nous choisissons la valeur 5 utilisée dans de précédentes études (LIN et al., 2009 ; LI et NENKOVA, 2014b). Nous optimisons le score de F_1 dans le cadre d'une classification binaire et le score de macro-F_1 en multiclasse. On utilise toujours la stratégie de pondération des instances pour gérer le déséquilibre des classes et ici également la divergence en termes de distribution sur les classes entre données naturelles et

	NatOnly		ArtOnly			
Données de test	Naturelles		Naturelles		Artificielles	
Paires de mots						
	P	F_1	P	F_1	P	F_1
Temporal	22,4	20,6	12,3	17,6	48,2	59,1
Contingency	42,4	50,2	31,0	25,2	31,7	40,9
Comparison	29,9	36,8	16,3	24,3	47,7	48,4
Expansion	63,3	56,8	59,5	53,7	54,1	49,2
base+lex						
Temporal	22,5	23,0	14,8	20,0	55,4	63,7
Contingency	40,9	50,9	37,1	27,1	28,5	41,0
Comparison	27,9	36,0	16,9	26,7	46,3	52,6
Expansion	66,0	61,6	60,0	52,0	57,7	63,5

Table 4.17.: Modèles de référence sur l'anglais en binaire au niveau 1 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, précision (« P ») et F_1 par relation.

artificielles. Dans le cas d'expériences comme l'entraînement sur les seules données artificielles et l'union des données, nous calculons les poids à partir de la distribution des données d'entraînement naturelles. La significativité est évaluée avec le test de Student apparié (t-test) et le test de Wilcoxon sur 20 sous-ensembles des données d'évaluation.

4.5.3.2 Résultats sur l'anglais en binaire au niveau 1

Modèles de référence

Nous présentons d'abord des résultats pour les systèmes de référence dans la table 4.17 pour la représentation uniquement fondée sur les paires de mots et la représentation augmentée de traits supplémentaires **base+lex**. Les résultats sont légèrement différents de ceux présentés dans le chapitre précédent pour l'entraînement et l'évaluation sur les données naturelles (NatOnly), différence due au fait que nous n'optimisons pas le seuil en fréquence. Cependant, ici notre visée est de tester l'effet des méthodes d'adaptation de domaine décrites précédemment et nous avons donc choisi de fixer ce paramètre supplémentaire.

On observe comme sur le français et dans les études précédentes une baisse de performance pour un modèle construit sur les données artificielles et évalué sur les données naturelles (ArtOnly avec évaluation sur les données naturelles) par rapport à un modèle directement construit sur les données naturelles disponibles (NatOnly). De plus, nous obtenons également des performances assez hautes pour les modèles entraînés et testés sur les données artificielles (ArtOnly avec évaluation sur les données artificielles). Les scores sont cependant bien inférieurs à ceux obtenus pour la tâche d'identification des relations explicites[10] ce qui montre l'importance de l'indice correspondant au connecteur.

Ici, contrairement à la configuration des expériences sur le français, les données artificielles proviennent du même corpus que les données naturelles et elles ne sont pas bruitées. Comme on dispose d'un nombre d'instances artificielles et naturelles équivalent, on devrait observer des résultats similaires si les données implicites et artificielles suivaient la même distribution. Ces résultats montrent qu'il existe bien des différences en termes distributionnels entre les deux types de données. Par ailleurs, la différence en termes de distribution sur les sorties étant gérée à l'aide d'une pondération des instances[11], on peut conclure que ces données sont différentes au niveau de la distribution des entrées et/ou du concept à apprendre. De plus, on observe que la baisse

10. Voir par exemple les scores obtenus à la constitution des données artificielles en section 4.4.2.2.
11. Les poids sont calculés sur les données naturelles.

est particulièrement forte pour les relations *Comparison* et *Contingency*, elle est beaucoup moins importante pour *Temporal* et *Expansion*. Ceci est en lien avec les remarques que nous avons faites au début de ce chapitre : les relations contrastives, apparaissant dans la classe *Comparison*, et causales, de la classe *Contingency*[12], pourraient correspondre plus souvent à des modifications lors de la suppression d'un connecteur. Les exemples artificiels de ces relations seraient donc logiquement moins similaires aux exemples naturels implicites. Ceci est en contradiction avec nos observations sur le français pour la relation *Contrast* mais cette relation était particulièrement sous-représentée pour le français, elle profitait donc particulièrement de l'ajout de nouvelles données.

L'utilisation de traits supplémentaires en plus des paires de mots (représentation **base+lex**) permet une amélioration des performances pour les modèles entraînés sur les données artificielles et testés sur les naturelles. Ce n'était pas forcément attendu car ces traits supplémentaires auraient pu être plus spécifiques à des relations non explicitées. Ces expériences montrent qu'au contraire les informations qu'ils encodent sont partagées par les deux types de données puisqu'ils permettent des améliorations, cependant assez légères, pour la majorité des modèles.

Modèles avec combinaisons de données

Nous limitons ici nos expériences aux systèmes de combinaison des données par union et par pondération des données naturelles, l'utilisation de sous-ensembles de données artificielles correspondant finalement à une stratégie similaire à la pondération mais avec un caractère aléatoire qui rend difficile toute généralisation. Concernant la combinaison des modèles, nous reprenons les expériences correspondant à l'ajout de traits, qui ont conduit aux meilleures performances sur le français, et celle effectuant une interpolation linéaire des modèles, l'initialisation d'un modèle par l'autre étant moins pertinent pour analyser l'influence de chaque type de données.

Contrairement aux expériences sur le français, nous optimisons ici les hyper-paramètres de l'algorithme présentés en section 3.4.3 correspondant à la norme de régularisation et à sa force. Pour l'interpolation linéaire, nous utilisons les meilleurs modèles construits sur chaque ensemble de données et optimisés sur l'ensemble de développement naturel. Pour l'ajout des traits de prédictions et éventuellement de probabilité, nous utilisons le meilleur modèle obtenu à partir des données artificielles puis nous optimisons les différents hyper-paramètres de l'algorithme. Pour la pondération des données naturelles, on teste ici moins de valeurs car on dispose de beaucoup moins de données artificielles, on duplique k fois les exemples avec $k \in \{2, 5, 10\}$.

Nous reportons dans le tableau 4.18 les scores pour les modèles combinant données ou modèles sans sélection d'exemples artificiels pour les représentations paires de mots et **base+lex**. Les méthodes de combinaison permettent en général d'obtenir des systèmes qui dépassent les scores obtenus en n'utilisant que les données artificielles à l'entraînement. De plus, les meilleurs scores obtenus améliorent également les résultats correspondant à un entraînement utilisant les seules données naturelles (NATONLY). Avec la représentation utilisant uniquement les paires de mots, on obtient ainsi des améliorations de $4,2\%$ de F_1 pour *Expansion*, de $3,2\%$ pour *Temporal*, de 2% pour *Comparison* et de $0,9\%$ pour *Contingency*. Avec la représentation **base+lex**, on obtient également des améliorations assez importantes pour *Temporal* (gain de 3% en F_1) et *Expansion* (gain de $4,2\%$) mais plus faible pour *Contingency* ($0,4\%$) et *Comparison* ($0,4\%$). Les améliorations obtenues ne sont cependant pas significatives.

12. Rappelons qu'il existe très peu d'exemples de type *Condition*, le second grand groupe de relation dans la classe *Contingency*, dans les données implicites ce qui nous a conduit à supprimer cette relation de notre ensemble. Il est effectivement très difficile d'exprimer une relation hypothétique sans connecteur.

	NatOnly	Union		NatW		AddPred		AddProb		LinInt	
Paires de mots											
	F_1	P	F_1	P	F_1	P	F_1	P	F_1	P	F_1
Temporal	20,6	22,7	**23,8**	15,8	23,6	23,3	21,9	21,0	20,0	19,1	22,2
Contingency	50,2	41,1	43,4	37,1	46,8	41,6	49,9	42,1	**51,1**	45,5	47,5
Comparison	36,8	28,1	30,0	26,0	35,9	30,3	**38,8**	30,0	38,6	27,0	34,9
Expansion	56,8	64,5	56,3	63,5	56,6	57,1	**61,0**	57,1	**61,0**	58,6	56,8
base+lex											
	F_1	P	F_1	P	F_1	P	F_1	P	F_1	P	F_1
Temporal	23,0	16,7	0,50	20,9	25,4	22,9	23,2	22,9	23,2	24,4	**26,0**
Contingency	50,9	43,9	46,8	41,4	50,4	40,9	50,9	40,9	**51,1**	44,9	50,7
Comparison	36,0	30,6	35,4	26,7	**36,4**	26,5	34,3	26,1	33,8	24,5	35,2
Expansion	61,6	66,2	60,1	65,0	58,1	65,5	61,2	65,8	61,2	55,5	**63,3**

Table 4.18.: Modèles avec combinaison sur l'anglais en binaire au niveau 1 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, précision (« P ») et F_1 par relation. Nous mettons en gras les meilleurs scores de F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons le score de F_1 de notre système de référence NatOnly pour faciliter la comparaison.

On observe que les meilleures méthodes de combinaison dépendent à la fois de la relation et de la représentation. L'ajout des traits de prédiction et, éventuellement, de probabilité, permet cependant d'obtenir les meilleurs scores dans la majorité des cas pour *Comparison* et *Contingency* ce qui rejoint les conclusions obtenues sur le français. La relation *Temporal* profite cependant moins de cette configuration que de l'ajout des données artificielles à l'ensemble d'entraînement (Union et NatW) ou d'une combinaison plus directe des modèles (LinInt), cette dernière configuration permettant également de bons scores pour *Expansion*. Par ailleurs, alors que l'union des données entraînait des baisses très importantes sur le français, les données naturelles étant noyées par rapport à la masse de données artificielles, ici ce système correspond encore à des scores relativement hauts et notamment pour *Temporal* et *Expansion*, ce qui est dû au fait que les deux types de données sont en quantité similaire. Ces observations rejoignent nos conclusions précédentes, à savoir que ces relations correspondent à des données artificielles plus similaires aux naturelles que pour les deux autres relations. Comme précédemment, la représentation **base+lex** permet généralement d'améliorer les performances par rapport aux modèles uniquement fondés sur les paires de mots. Notons cependant que les paires de mots permettent déjà d'obtenir des scores assez hauts, et même meilleurs pour la relation *Comparison* montrant l'importance de l'information lexicale. Ces paires de mots sont en général uniquement celles des exemples implicites, puisque les meilleurs modèles correspondent souvent à l'ajout de traits de prédiction à des modèles entraînés sur les seules données naturelles, à l'exception de *Temporal*.

Modèles avec sélection d'exemples

Nous rapportons dans le tableau 4.19 les scores obtenus pour les modèles utilisant la sélection d'exemples. Nous testons ici des seuils de probabilité $s \in \{0,3\ , 0,9\}$ avec un incrément de $0,1$. Pour rappel, cette configuration nous avait permis sur le français d'obtenir des améliorations significatives par rapport au modèle entraîné sur les seules données naturelles. Ici, cependant, nous obtenons des scores similaires, voire légèrement inférieurs, à ceux obtenus sans sélection d'exemples. La seule exception est *Expansion* pour laquelle nous obtenons une amélioration significative de $5,6\%$ en F_1 par rapport au modèle entraîné sur les seules données naturelles disponibles, avec cependant une large baisse de la précision, le modèle prédisant en fait une large majorité des exemples vers cette relation.

	NatOnly	Union		NatW		AddPred		AddProb		LinInt	
Paires de mots											
	F_1	P	F_1	P	F_1	P	F_1	P	F_1	P	F_1
Temporal	20,6	25,4	23,6	15,9	**23,9**	24,3	17,1	21,0	19,1	19,2	20,5
Contingency	50,2	38,7	45,1	36,3	46,6	41,7	49,6	41,5	**50,1**	43,3	50,0
Comparison	36,8	26,0	36,3	31,3	**37,5**	29,6	37,0	30,0	37,1	23,3	34,3
Expansion	56,8	62,5	59,6	63,8	58,5	56,2	58,4	56,2	58,4	55,1	**66,7***
base+lex											
	F_1	P	F_1	P	F_1	P	F_1	P	F_1	P	F_1
Temporal	23,0	17,4	20,0	20,9	**25,4**	23,3	24,1	22,7	23,8	21,3	22,4
Contingency	50,9	43,4	50,5	31,3	41,3	40,8	50,8	41,3	**51,6**	44,8	50,7
Comparison	36,0	29,8	**38,6**	27,9	37,2	27,3	35,5	27,3	35,5	26,5	35,2
Expansion	61,6	65,4	61,2	63,0	60,1	65,2	60,4	65,2	60,6	54,0	**67,2***

Table 4.19.: Modèles avec sélection sur l'anglais en binaire au niveau 1 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, précision (« P ») et F_1 par relation. Nous mettons en gras les meilleurs scores de F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons le score de F_1 de notre système de référence NatOnly pour faciliter la comparaison. * $p \leq 0.1$ par rapport à NatOnly.

	NatOnly		ArtOnly		ArtOnly+selec	
Paires de mots						
	P	F_1	P	F_1	P	F_1
Temporal	22,4	20,6	12,3	17,6	11,8	17,9
Contingency	42,4	50,2	31,0	25,2	38,8	40,8
Comparison	29,9	36,8	16,3	24,3	22,4	37,0
Expansion	63,3	56,8	59,5	53,7	56,1	53,6
base+lex						
	P	F_1	P	F_1	P	F_1
Temporal	22,5	23,0	14,8	20,0	14,8	20,1
Contingency	40,9	50,9	37,1	27,1	41,2	35,7
Comparison	27,9	36,0	16,9	26,7	18,6	24,6
Expansion	66,0	61,6	60,0	52,0	58,3	57,0

Table 4.20.: Modèles de référence et comparaison avec le modèle entraîné sur les données artificielles et évaluées sur les naturelles avec sélection (ArtOnly+selec) sur l'anglais en binaire au niveau 1 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, précision (« P ») et F_1 par relation.

Le fait que la sélection n'améliore pas les scores peut provenir du fait qu'elle est essentiellement destinée à gérer le bruit dans les données artificielles, or il n'y a pas de bruit dans les données issues du PDTB. Ceci pourrait donc signifier que la méthode ne gère pas en plus le problème de redondance : idéalement, on voudrait ne conserver que les exemples artificiels pour lesquels le connecteur est redondant, donc pour lesquels le modèle artificiel parvient à une bonne identification malgré l'absence du connecteur. Cependant, nous avons observé que la sélection permettait d'améliorer généralement les performances du modèle entraîné sur les données artificielles et testé sur les données naturelles. On a ainsi au mieux des gains allant jusqu'à 15% de F_1 avec la représentation sur les paires de mots et de presque 9% avec la représentation **base+lex**, les gains étant essentiellement observables pour les relations *Contingency* et *Comparison*, cette dernière avec la représentation sur les paires de mots, comme on peut le voir dans le tableau 4.20.

La sélection permet donc bien d'améliorer la généralisation que l'on peut obtenir à partir des données artificielles sur les données naturelles. Le fait que cette amélioration ne se répercute pas sur les méthodes de combinaison peut être dû au fait que l'on dispose d'assez peu de données artificielles, surtout par rapport au nombre de données naturelles. En effet, pour le français, le manque de données était bien plus important, nous disposions d'un ensemble d'environ 200

	NatOnly	ArtOnly	
Données de test	Naturelles	Naturelles	Artificielles
Paires de mots			
Macro-F_1	39,9	29,2	48,2
Macro-prec	41,7	30,6	48,2
Exactitude	52,7	36,9	49,6
base+lex			
Macro-F_1	41,2	30,5	53,4
Macro-prec	41,9	32,9	52,9
Exactitude	50,6	37,9	54,4

Table 4.21.: Modèles de référence sur l'anglais en multiclasse au niveau 1 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »)), et scores d'exactitude de chaque système.

instances naturelles contre plus de 10 000 pour l'anglais. Ces résultats semblent donc indiquer que les méthodes utilisées nécessitent une grande quantité de données artificielles. Il est possible que des méthodes d'adaptation de domaine plus sophistiquées soient capables de profiter d'un apport même minimal de nouvelles données. Notons que, malgré tout, la sélection d'exemples semble ici aussi cruciale, elle permet par exemple de construire des systèmes à partir des seules données artificielles correspondant à des résultats qui ne sont pas très inférieurs à ceux obtenus avec les données naturelles, sauf pour la relation *Contingency*.

Finalement, nous obtenons au mieux 26,0% de F_1 pour *Temporal* (LinInt), 51,6% pour *Contingency* (AddProb+selec), 38,8% pour *Comparison* (AddPred) et 67,2% pour *Expansion* (LinInt+selec). Par comparaison, Rutherford et Xue (2015) rapportent pour leur meilleur système correspondant à une union des données avec une méthode de sélection plus complexe 33,3% de F_1 pour *Temporal*, 53,8% pour *Contingency*, 41,0% pour *Comparison* et 69,1% pour *Expansion*. Nous obtenons donc des scores inférieurs pour toutes les relations ce qui montre soit l'importance de disposer d'un nombre plus important de données artificielles soit la pertinence d'une méthode de sélection plus sophistiquée.

4.5.3.3 Résultats sur l'anglais en multiclasse au niveau 1

Nous présentons dans cette section les résultats pour les expériences menées en multiclasse au niveau 1 de sens sur le PDTB. La configuration est la même qu'en binaire sauf que nous optimisons le score de macro-F_1 [13].

Modèles de référence

Nous présentons dans le tableau 4.21 les résultats pour les systèmes de référence. Notons qu'ici aussi les résultats de NatOnly sont différents des scores présentés dans le chapitre précédent (section 3.4.3). Les observations sont les mêmes que précédemment : une baisse des performances pour les modèles entraînés et évalués sur des types de donnée différents révélant les différences en termes distributionnels avec cependant une baisse en termes de macro-F_1 moins importante que ce que l'on avait pu observer sur le français probablement dû à une similarité plus forte entre les deux types de données comme indiqué dans le cas binaire.

13. Sur le français nous optimisions la micro-exactitude mais les données étaient globalement équilibrées. Les données du PDTB sont fortement déséquilibrées, il est donc plus naturel d'optimiser une macro-moyenne.

	NatOnly	Union	NatW	AddPred	AddProb	LinInt
Paires de mots						
Macro-F_1	39,9	39,3	**41,2**	40,1	39,6	38,8
Macro-prec	41,7	39,3	41,6	42,3	41,5	41,7
Exactitude	52,7	49,1	50,8	53,2	53,0	53,4
base+lex						
Macro-F_1	41,2	40,1	42,2	42,8	**43,8**	41,0
Macro-prec	41,9	40,3	41,3	43,1	43,8	42,4
Exactitude	50,6	50,0	49,8	51,1	51,8	53,0

Table 4.22.: Modèles avec combinaison sur l'anglais en multiclasse au niveau 1 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système. Nous mettons en gras les meilleurs scores de macro-F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons les scores de notre système de référence NatOnly pour faciliter la comparaison.

Modèles avec combinaisons de données

Lorsque nous combinons les données ou les modèles nous obtenons des systèmes correspondant à des scores de F_1 supérieurs à ceux obtenus avec le système de référence NatOnly comme on peut le voir dans le tableau 4.22. Le meilleur système correspond à une macro-F_1 de $43,8\%$ soit une amélioration de $2,6\%$ par rapport à notre système de référence. Notons également que ce score est supérieur au meilleur système présenté dans le chapitre précédent, soit celui fondé sur la représentation **base+lex+synt** qui correspond à un score de macro-F_1 de $42,4\%$. De manière assez similaire à ce que nous avions obtenu pour le français, c'est le système fondé sur l'ajout de traits, ici prédiction et probabilité, qui permet d'obtenir les meilleurs scores.

(a) AddProb.

Rel	P	R	F_1
Temp	22,0	38,2	**28,0**
Cont	44,9	46,4	**45,6**
Comp	44,4	35,6	**39,5**
Expa	63,9	60,4	62,1

(b) (Rutherford et Xue, 2015).

Rel	P	R	F_1
Temp	38,5	9,1	14,7
Cont	49,3	39,6	43,9
Comp	44,9	27,6	34,2
Expa	61,4	78,8	**69,1**

Table 4.23.: Scores de précision (« P »), rappel (« R ») et F_1 par relation pour le meilleur système fondé sur la combinaison des données naturelles et artificielles AddProb avec la représentation **base+lex**. Nous reprenons également les scores du meilleur système présenté dans (Rutherford et Xue, 2015).

Dans la dernière étude en multiclasse au niveau 1, Rutherford et Xue (2015) rapportent un score de macro-F_1 de $40,5\%$ et un score d'exactitude de $57,1\%$. Notons que ces auteurs optimisent l'exactitude et non la macro-F_1, il est donc difficile de comparer ces scores globaux. Nous rapportons les scores par classe pour notre meilleur système dans le tableau 4.23a ainsi que les meilleurs scores actuels sur cette tâche présentés dans (Rutherford et Xue, 2015) dans le tableau 4.23b. Pour rappel, ces auteurs utilisent également des données artificielles avec une méthode de sélection fondée sur des mesures de divergence entre les deux types de données. Comme le montrent les tableaux, nous obtenons des résultats supérieurs à l'état de l'art pour trois des quatre relations. Pour la relation *Expansion*, le système de Rutherford et Xue (2015) profite probablement de l'utilisation de traits de coréférence.

	NatOnly	Union	NatW	AddPred	AddProb	LinInt
Paires de mots						
Macro-F_1	39,9	40,4	**41,5**	40,2	40,3	38,8
Macro-prec	41,7	40,5	41,7	42,3	42,3	41,7
Exactitude	52,7	50,6	51,6	53,4	53,4	53,4
base+lex						
Macro-F_1	41,2	42,2	**43,8**	43,4	43,5	41,9
Macro-prec	41,9	43,3	43,0	43,6	43,7	42,6
Exactitude	50,6	53,0	51,6	51,4	51,7	53,4

Table 4.24.: Modèles avec sélection sur l'anglais en multiclasse au niveau 1 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système. Nous mettons en gras les meilleurs scores de macro-F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons les scores de notre système de référence NatOnly pour faciliter la comparaison.

Modèles avec sélection d'exemples

Nous présentons dans le tableau 4.24 les résultats pour les systèmes utilisant notre stratégie de sélection d'exemples. Une fois encore, tous les systèmes correspondent à des performances supérieures à celles obtenues avec le système de référence (NatOnly). Cette fois-ci cependant, les meilleurs systèmes sont obtenus avec la pondération des exemples naturels. Notons que c'était également le cas avec la représentation utilisant uniquement les paires de mots sans sélection d'exemples. L'ajout des traits de prédiction et éventuellement de probabilité permet également d'obtenir des scores assez hauts. Ceci étant dit la sélection ne permet pas de dépasser les scores obtenus sans sélection ce qui rejoint les conclusions de nos expériences menées en binaire présentées dans la section précédente.

4.5.3.4 Résultats sur l'anglais en multiclasse au niveau 2

Nous testons enfin des systèmes en multiclasse au niveau 2 de sens de la hiérarchie du PDTB. Ce niveau correspond à 11 relations de grain assez fin, du même type que celles que nous avions étudiées en français. Nous présentons les scores de référence obtenus pour ce niveau dans le tableau 4.25, les scores sans sélection dans le tableau 4.27 et ceux avec sélection dans le tableau 4.28.

On observe la même tendance que pour le niveau 1 de sens : une chute importante des performances sur les données naturelles entre un modèle entraîné sur des données du même type et un modèle construit sur les artificielles. Par contre ici l'utilisation de l'ensemble de traits **base+lex** conduit à des baisses de performances quelque soit le modèle. Ceci ne peut être dû qu'au fait que nous n'avons pas optimisé le filtre de fréquence, seul paramètre modifié par rapport aux systèmes de référence présentés dans le chapitre précédent en section 3.4.3.3. Ceci montre l'importance de cet hyper-paramètre gérant brutalement mais apparemment d'une manière assez efficace le problème d'éparpillement.

Une autre différence par rapport au niveau 1 est le fait que le modèle entraîné et évalué sur les données artificielles (ArtOnly évalué sur les artificielles) correspond à des performances très basses. Au niveau 1 ce modèle correspond à une macro-F_1 de presque 50%, bien supérieure aux 30% obtenus par le modèle entraîné et testé sur les données naturelles. Ici, le modèle entraîné et testé sur les données artificielles correspond à un score similaire à celui du modèle naturel (NatOnly). Nous avons vu que les performances pour les explicites en prenant en compte le connecteur dans la modélisation étaient inférieures au niveau 2 par rapport au niveau 1. Elles restaient quand même de l'ordre de 64% de macro-F_1(voir section 4.4.2.2, tableau 4.10) contre

	NatOnly	ArtOnly	
Données de test	Naturelles	Naturelles	Artificielles
Paires de mots			
Macro-F_1	23,5	16,3	24,4
Macro-prec	35,6	20,7	33,8
Exactitude	39,9	23,7	40,4
base+lex			
Macro-F_1	23,6	13,0	24,6
Macro-prec	23,4	18,2	24,4
Exactitude	36,5	20,5	41,8

Table 4.25.: Modèles de référence sur l'anglais en multiclasse au niveau 2 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »)), et scores d'exactitude de chaque système.

seulement 24,4% ici, le problème ne vient donc pas seulement de l'augmentation du nombre de classes, et donc du déséquilibre entre les classes plus fort au niveau 2. Cette baisse peut indiquer que la modélisation des données utilisée ici n'est pas suffisamment riche pour les distinctions fines opérées au niveau 2. Comme nous évaluons ici deux modélisations et qu'elles correspondent à des performances similaires, nous pensons que le problème est plus complexe. Il se pourrait qu'à ce niveau de grain fin, le connecteur ait une importance plus cruciale donc que la redondance du connecteur dépende du grain d'analyse : bien que l'on puisse par exemple distinguer relativement facilement une instance de la classe *Temporal* ou *Expansion* sans le connecteur, reconnaître une distinction entre une relation de synchronie ou d'asynchronie ou faire la différence entre un lien de type *Conjunction* ou *Alternative* pourrait être plus difficile.

(a) Modèle entraîné et évalué sur les données explicites (jeu de traits décrit en section 4.4.2.2).

Niv 1	Rel niv 2	P	F_1
Temp	*Async*	94,3	88,3
	Sync	63,0	73,7
Cont	*Cause*	98,0	92,9
	Prag Cause	0,0	0,0
Comp	*Contr*	93,1	91,6
	Conc	34,5	38,5
Exp	*Conj*	85,6	91,2
	Inst	100,0	100,0
	Rest	100,0	25,0
	Alt	88,9	94,1
	List	50,0	6,5

(b) Modèle entraîné et évalué sur les données artificielles (paires de mots).

Niv 1	Rel niv 2	P	F_1
Temp	*Async*	35,1	43,2
	Sync	33,5	40,0
Cont	*Cause*	28,8	30,6
	Prag Cause	0,0	0,0
Comp	*Contr*	42,1	38,9
	Conc	14,8	8,2
Exp	*Conj*	54,5	50,3
	Inst	38,5	28,6
	Rest	100,0	16,7
	Alt	25,0	11,4
	List	0,0	0,0

(c) Modèle entraîné et évalué sur les données implicites naturelles (paires de mots).

Niv 1	Rel niv 2	P	F_1
Temp	*Async*	29,7	24,2
	Sync	100,0	13,3
Cont	*Cause*	40,0	46,2
	Prag Cause	0,0	0,0
Comp	*Contr*	32,3	32,6
	Conc	50,0	10,5
Exp	*Conj*	43,1	42,0
	Inst	59,6	54,4
	Rest	36,5	35,3
	Alt	0,0	0,0
	List	0,0	0,0

Table 4.26.: Scores de précision (« P ») et F_1 par relation pour un modèle d'identification des relations explicites, artificielles et implicites. au niveau 2. Nous séparons les relations en sous-groupes correspondant aux relations de niveau 1 (« *Temp* » pour *Temporal*, « *Cont* » pour *Contingency*, « *Comp* » pour *Comparison* et « *Exp* » pour *Expansion*).

En comparant les scores par classe pour deux représentations des relations explicites, soit en prenant en compte le connecteur (tableau 4.26a) soit sans le considérer ce connecteur dans la modélisation (tableau 4.26b), on observe que les relations de la classe *Temporal* semblent moins affectées par la non prise en compte du connecteur, les scores sont divisés par 2 mais on conserve des scores du même ordre pour les deux relations composant cette classe. Les relations *Asynchrony* et *Synchrony* restent donc relativement bien identifiées, la redondance du connecteur est assez forte. Par contre, dans la classe *Comparison*, le score pour *Contrast* est divisé par 2,3 et celui de *Concession* presque par 5. La différence entre ces deux dernières relations est assez fine, rappelons qu'elles sont regroupées en une seule relation dans le cadre de la SDRT. Il est clair qu'ici le connecteur est le meilleur indice de cette distinction. On observe le même type de différences au sein de la classe *Expansion*. En ce qui concerne la relation *Cause*, on a par contre une baisse que l'on ne peut pas expliquer de la même façon au sens où de toute façon c'est la seule relation de la

	NatOnly	Union	NatW	AddPred	AddProb	LinInt
Paires de mots						
Macro-F_1	**23,5**	20,7	21,5	23,2	23,2	22,5
Macro-prec	35,6	22,8	21,9	35,2	35,1	34,7
Exactitude	39,9	35,1	34,9	39,6	39,6	39,4
base+lex						
Macro-F_1	23,6	21,9	23,0	22,5	23,0	**23,7**
Macro-prec	23,4	23,5	22,7	25,0	22,9	24,3
Exactitude	36,5	34,2	35,0	36,3	36,7	37,0

Table 4.27.: Modèles avec combinaison sur l'anglais en multiclasse au niveau 2 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »)), et scores d'exactitude de chaque système.

	NatOnly	Union	NatW	AddPred	AddProb	LinInt
Paires de mots						
Macro-F_1	23,5	22,2	**23,9**	23,3	22,8	22,2
Macro-prec	35,6	22,5	24,3	30,9	33,3	34,6
Exactitude	39,9	36,6	38,2	39,8	39,0	38,7
base+lex						
Macro-F_1	23,6	23,0	24,0	**24,1**	23,7	23,0
Macro-prec	23,4	24,0	24,1	27,3	23,6	24,7
Exactitude	36,5	37,7	35,7	36,7	36,8	36,1

Table 4.28.: Modèles avec sélection sur l'anglais en multiclasse au niveau 2 : corpus artificiel constitué à partir du PDTB et représentations paires de mots et base+lex, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »)), et scores d'exactitude de chaque système.

classe *Contingency* à correspondre à un score non nul. Cependant le score pour cette relation est également très inférieur à celui obtenu pour un modèle entraîné et évalué sur les données naturelles (tableau 4.26c). Ceci indique probablement que le connecteur est un indice particulièrement crucial pour cette relation et que les implicites mettent en œuvre des indices différents.

Bien qu'obtenir de bonnes performances pour le modèle artificiel n'ait que peu d'intérêt de manière générale, cela peut cependant correspondre à une condition nécessaire à la réussite des stratégies d'adaptation puisque, comme nous venons de le dire, cela semble indiquer que l'hypothèse de redondance du connecteur est non vérifiée. Il se peut que ces nouvelles données apportent malgré tout des informations supplémentaires mais, étant donné que l'on a déjà de meilleurs scores en utilisant les seules données implicites naturelles pour certaines relations, comme *Cause*, *Instantiation* et *Restatement*, il est peu probable qu'elles conduisent à des améliorations.

Cela semble en effet se vérifier. Si les stratégies mises en œuvre permettent de dépasser largement le modèle entraîné sur les données artificielles, on n'observe toutefois aucune amélioration pour les systèmes combinant les données ou les modèles par rapport à l'entraînement sur les seules données naturelles (tableau 4.27). Nous pensons donc qu'ici le problème vient du fait que les informations restantes dans les arguments des données artificielles ne sont pas suffisantes pour identifier la relation, donc que ces nouvelles données n'apportent pas d'informations pertinentes. Nous envisageons donc de mener d'autres expériences pour vérifier s'il existe une représentation pour laquelle on puisse obtenir des améliorations. Il se peut également que le problème vienne du fait que l'on dispose d'assez peu de données artificielles surtout pour certaines classes.

La sélection permet généralement d'améliorer les scores obtenus sans sélection (tableau 4.28), amélioration cependant assez légère : si nous dépassons cette fois les performances du modèle entraîné et évalué sur les données naturelles, les améliorations en termes de F_1 restent basses,

	NatOnly		ArtOnly			
Données de test	Naturelles		Naturelles		Artificielles	
Paires de mots						
	P	F_1	P	F_1	P	F_1
Temporal	22,4	20,6	6,1	9,4	52,4	60,0
Contingency	42,4	50,2	37,2	32,8	55,3	62,0
Comparison	29,9	36,8	15,5	24,1	48,5	57,8
Expansion	63,3	56,8	54,2	52,3	59,2	62,7

Table 4.29.: Modèles de référence sur l'anglais en binaire au niveau 1 : corpus artificiel constitué à partir du *Bllip* et représentations paires de mots, précision (« P ») et F_1 par relation.

0,4% avec la représentation sur les paires de mots et 0,5% avec **base+lex**, et non significatives. Par ailleurs, la sélection n'améliore pas vraiment les scores obtenus pour le modèle entraîné et évalué sur les données artificielles contrairement à ce que l'on avait observé en multiclasse au niveau 1 ce qui semble indiquer que le problème vient bien du type de données ajoutées.

4.5.4 Expériences sur l'anglais, corpus artificiel *Bllip*

On présente dans cette section les résultats obtenus en utilisant les données artificielles construites automatiquement à partir du *Bllip*. Nous ne testons ici que la représentation sur les paires de mots. La configuration est la même que précédemment sauf que l'on fixe un filtre de seuil plus élevé pour des raisons computationnelles : les données artificielles sont filtrées avec un seuil de 100 et les données naturelles de 20. A l'origine, les données artificielles *Bllip* contiennent plus de 2 millions de traits pour environ 1 million d'exemples. Nous réduisons l'espace à environ 500 000 dimensions. Pour le même type de raison, nous ne reprenons pas l'expérience consistant à dupliquer les exemples naturels dans l'ensemble d'entraînement.

4.5.4.1 Résultats sur l'anglais en binaire au niveau 1

Nous présentons dans le tableau 4.29 les scores de référence correspondant aux modèles entraînés sur un seul type de données. On observe notamment que le modèle entraîné et évalué sur les données artificielles correspond à des performances supérieures à celles obtenues avec le corpus artificiel manuel notamment pour *Contingency* (gain de 21,1% en F_1) et *Comparison* (gain de 9,4%). Le fait de disposer d'une plus grande quantité de données ne suffit cependant pas à obtenir des scores supérieurs pour le modèle entraîné sur les données artificielles et évaluées sur les données naturelles : on obtient une baisse importante pour *Temporal* (baisse de 8,2% en F_1) et un score similaire pour *Comparison*. Par contre, on a une amélioration pour *Contingency* (gain de 7,6%). On a donc ici un apport potentiel dû à la large quantité de données ajoutées et une dégradation possible à cause du bruit.

Les modèles avec combinaison de données permettent d'obtenir des améliorations par rapport au modèle entraîné sur les seules données naturelles (tableau 4.30) notamment pour *Temporal* pour laquelle on obtient un score supérieur à celui obtenu avec le corpus artificiel manuel avec le même jeu de traits (23,8% de F_1). Pour les autres relations, les améliorations par rapport au modèle NatOnly sont cependant inférieures à celles obtenues en utilisant le corpus artificiel construit à partir du PDTB. On observe également une amélioration pour *Expansion* en termes de F_1 : on obtient ici un score de 65,1% contre 61,0% avec le corpus manuel. On a cependant pour cette relation une perte en termes de précision. Pour rappel nous avions obtenu au mieux 51,1% de F_1 pour *Contingency* contre 50,5% ici et 38,8% pour *Comparison* contre 37,9% ici. Il est intéressant de noter que les résultats obtenus avec le modèle correspondant à une simple union des

données correspond à des résultats inférieurs, voire très inférieurs à ceux obtenus avec le corpus artificiel manuel. Cette baisse montre que le bruit induit par l'annotation automatique ainsi que potentiellement la différence en termes de domaine on un impact négatif sur le modèle malgré la masse de données ajoutée.

	NatOnly	Union		AddPred		AddProb		LinInt	
	Paires de mots								
	F_1	P	F_1	P	F_1	P	F_1	P	F_1
Temporal	$20,6$	$9,8$	$15,4$	$18,6$	$22,3$	$23,4$	$\mathbf{24,8}$	$5,5$	$9,0$
Contingency	$50,2$	$39,8$	$39,6$	$42,0$	$\mathbf{50,5}$	$40,9$	$50,1$	$40,4$	$47,0$
Comparison	$36,8$	$16,3$	$25,2$	$28,0$	$34,8$	$27,7$	$34,4$	$34,7$	$\mathbf{37,9}$
Expansion	$56,8$	$56,7$	$55,8$	$56,3$	$61,9$	$56,3$	$61,9$	$54,9$	$\mathbf{65,1}$

Table 4.30.: Modèles avec combinaison sur l'anglais en binaire au niveau 1 : corpus artificiel constitué à partir du *Bllip* et représentations paires de mots, précision (« P ») et F_1 par relation. Nous mettons en gras les meilleurs scores de F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons le score de F_1 de notre système de référence NatOnly pour faciliter la comparaison.

	NatOnly	Union		AddPred		AddProb		LinInt	
	Paires de mots								
	F_1	P	F_1	P	F_1	P	F_1	P	F_1
Temporal	$20,6$	$16,5$	$17,7$	$19,7$	$18,6$	$20,8$	$\mathbf{21,4}$	$20,0$	$21,0$
Contingency	$50,2$	$38,2$	$39,4$	$42,3$	$50,4$	$42,5$	$\mathbf{51,5}$	$40,1$	$46,4$
Comparison	$\mathbf{36,8}$	$20,2$	$26,5$	$27,1$	$36,2$	$27,0$	$35,9$	$30,1$	$35,5$
Expansion	$56,8$	$56,2$	$59,7$	$56,2$	$61,6$	$56,2$	$61,6$	$55,1$	$\mathbf{66,1}^*$

Table 4.31.: Modèles avec sélection sur l'anglais en binaire au niveau 1 : corpus artificiel constitué à partir du *Bllip* et représentations paires de mots, précision (« P ») et F_1 par relation. Nous mettons en gras les meilleurs scores de F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons le score de F_1 de notre système de référence NatOnly pour faciliter la comparaison. * $p \leq 0.1$ par rapport à NatOnly.

4.5.4.2 Résultats sur l'anglais en multiclasse

En multiclasse au niveau 1 de relation, l'utilisation du corpus artificiel constitué à partir du *Bllip* permet d'obtenir un modèle artificiel bien plus performant sur les données artificielles que lorsqu'il était construit à partir des seules données explicites du PDTB : le score de macro-F_1 passe ainsi de $48,2\%$, en utilisant le PDTB, à $60,1\%$, en utilisant le *Bllip*. Les scores de référence sont repris dans le tableau 4.32. Par contre, ce modèle artificiel est moins performant sur les données naturelles. Rappelons que les données artificielles issues du PDTB ne sont pas bruitées, et qu'elles correspondent au même domaine que les données naturelles implicites : ici, l'ajout de bruit et la différence de domaine expliquent cette baisse de performance d'environ 3% en macro-F_1. La baisse est du même ordre que ce que nous avons observé en binaire : les scores par relation sont en fait généralement inférieurs aux scores de F_1 des systèmes binaires, ce qui est à mettre en lien avec la plus grande difficulté de la classification multiclasse.

	NatOnly	ArtOnly	
Données de test	Naturelles	Naturelles	Artificielles
	Paires de mots		
Macro-F_1	$39,9$	$26,5$	$60,1$
Macro-prec	$41,7$	$28,1$	$60,2$
Exactitude	$52,7$	$34,1$	$60,1$

Table 4.32.: Modèles de référence sur l'anglais en multiclasse au niveau 1 : corpus artificiel constitué à partir du *Bllip* et représentations paires de mots, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système.

L'utilisation des stratégies de combinaison ne permet pas d'aboutir à une amélioration des performances par rapport au modèle entraîné sur les seules données naturelles (NATONLY) contrairement à ce que nous avons obtenu avec les données artificielles constituées à partir du PDTB et sur le français. Pour rappel, le meilleur modèle, avec le PDTB, correspondait à un score de macro-F_1 de $41,2\%$. Les résultats avec les données artificielles *Bllip* sont repris dans le tableau 4.33. De plus, en binaire, nous avons également obtenu des améliorations avec les données artificielles issues du *Bllip*. Ici, le déséquilibre des classes joue un rôle important, les systèmes ont tendance à prédire la relation *Expansion* par défaut, malgré l'utilisation d'une stratégie visant à gérer le déséquilibre des classes. Il faudra donc envisager la mise en place d'une autre méthode pour prendre en compte ce déséquilibre. On note que l'union des données naturelles et artificielles conduit à un score bien inférieur à celui obtenu en utilisant les données artificielles PDTB : la baisse entraînée par l'utilisation de ces données bruitées observée avec le modèle artificiel se répercute sur le modèle par union, puisque les données naturelles sont ici bien moins nombreuses que les données naturelles (environ 1 instance naturelle pour 100 instances artificielles). On a donc le même effet d'écrasement que ce que l'on a observé dans les expériences sur le français.

	NATONLY	UNION	ADDPRED	ADDPROB	LININT
Paires de mots					
Macro-F_1	$\mathbf{39,9}$	$31,7$	$39,0$	$\mathbf{39,9}$	$37,4$
Macro-prec	$41,7$	$32,8$	$40,3$	$41,1$	$39,7$
Exactitude	$52,7$	$39,3$	$51,6$	$52,1$	$51,8$

Table 4.33.: Modèles avec combinaison sur l'anglais en multiclasse au niveau 1 : corpus artificiel constitué à partir du *Bllip* et représentations paires de mots, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système. Nous mettons en gras les meilleurs scores de macro-F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons les scores de notre système de référence NATONLY pour faciliter la comparaison.

Nous reportons dans le tableau 4.34 les scores obtenus lorsque nous utilisons la méthode de sélection des exemples artificiels. Cette sélection a un impact important, puisque le modèle correspondant à l'union des deux types de données correspond à une amélioration de plus de 6% de macro-F_1. La sélection permet donc bien de conserver des exemples artificiels plus pertinents pour la tâche de classification des exemples naturels implicites. Cependant, la combinaison entre données naturelles et artificielles est cruciale : le modèle artificiel entraîné sur les seules données sélectionnées ne permet pas de dépasser les 30% de macro-F_1 lorsqu'il est évalué sur les données naturelles. Finalement, les scores obtenus pour tous les modèles restent inférieurs ou similaires à ceux du modèle entraîné sur les données naturelles (NATONLY).

Les expériences en multiclasse au niveau 2 conduisent aux mêmes conclusions, le meilleur modèle est obtenu avec le modèle ADDPROB mais les scores obtenus sont inférieurs à ceux du modèle entraîné sur les seules données naturelles (macro-F_1 de $21,0\%$ et exactitude de $34,6\%$ contre

	NATONLY	UNION	ADDPRED	ADDPROB	LININT
Paires de mots					
Macro-F_1	$\mathbf{39,9}$	$37,9$	$39,1$	$\mathbf{39,9}$	$37,4$
Macro-prec	$41,7$	$43,9$	$40,5$	$41,2$	$40,8$
Exactitude	$52,7$	$52,7$	$52,3$	$52,8$	$53,0$

Table 4.34.: Modèles avec sélection sur l'anglais en multiclasse au niveau 1 : corpus artificiel constitué à partir du *Bllip* et représentations paires de mots, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système. Nous mettons en gras les meilleurs scores de macro-F_1 obtenus pour chaque représentation et nous soulignons le meilleur score global. Nous rappelons les scores de notre système de référence NATONLY pour faciliter la comparaison.

respectivement $23,5\%$ et $39,9\%$ avec NATONLY), avec une dégradation pour toutes les relations en termes de F_1, à l'exception d'une légère amélioration pour *Asynchronous* (F_1 de $25,5\%$ avec ADDPROB contre $24,2\%$ avec NATONLY). Nous remarquons également que les scores sont inférieurs à ce que nous avons obtenu en utilisant le corpus artificiel construit à partir du PDTB, par exemple, la simple union des données conduit à seulement $13,8\%$ de macro-F_1 (contre $20,7\%$ avec le PDTB artificiel), ce qui est similaire à ce que nous avons noté pour le niveau 1.

Dans de futures expériences, nous envisageons d'évaluer des méthodes supplémentaires de combinaison des données en utilisant les données artificielles sélectionnées à partir du *Bllip*, et tout d'abord la méthode de pondération des données naturelles. Ensuite, nous voulons évaluer des méthodes de type *boosting* (FREUND et SCHAPIRE, 1997), également utilisées dan un cadre d'adaptation de domaine (DAI et al., 2007), et qui visent à identifier les instances à pondérer plus fortement, parce que plus difficiles à classer, et les instances à pondérer plus faiblement, parce qu'elles correspondent à des exemples naturels plus faciles à classer ou à des exemples artificiels engendrant une baisse des scores, donc bruités ou trop divergents.

4.6 Conclusion du chapitre

Les expériences présentées dans ce chapitre mettaient en œuvre des méthodes d'adaptation de domaine destinées à gérer les différences distributionnelles entre données naturelles et artificielles. Nous avons obtenu sur le français des améliorations significatives pour un système multiclasse montrant que le manque de données, particulièrement important pour cette langue, pouvait être géré par l'apport massif de données annotées automatiquement. Le meilleur système correspond à un score d'exactitude de $41,7\%$ pour quatre relations (*Contrast*, *Explanation*, *Continuation*, *Result*).

Nous avons également obtenu des améliorations sur le corpus anglais du PDTB, améliorations particulièrement importantes au niveau 1. Pour le niveau 2, nous avons conclu que le manque de succès relatif de notre stratégie résidait dans le fait que la suppression du connecteur avait un impact plus fort, cet indice permettant notamment de faire des distinctions plus fines comme celle qui existe entre *Contrast* et *Concession*. Nos expériences montrent également que la stratégie de sélection d'exemples n'est pas très efficace lorsque nous utilisons un corpus artificiel manuel, probablement parce que ces données sont assez peu nombreuses. Cependant, même avec les données artificielles issues du *Bllip*, les améliorations apportées par la sélection sont assez fluctuantes : nous obtenons des améliorations pour certaines relations, par exemple en binaire au niveau 1 les relations *Contingency* et *Expansion* profitent de la sélection mais pas *Temporal* et *Comparison*, et/ou pour certaines méthodes, la méthode par union est généralement améliorée par la sélection, c'est moins souvent le cas pour les autres méthodes de combinaison.

En binaire au niveau 1, les stratégies de combinaison nous ont permis d'obtenir des scores dépassant nos systèmes de référence pour *Temporal* ($26,0\%$ de F_1), *Comparison* ($38,8\%$) et *Expansion* ($67,2\%$). Pour *Contingency*, le système de référence permet d'obtenir un meilleur score de F_1 ($54,1\%$ contre $51,6\%$ au mieux avec combinaison). Cela montre peut-être l'importance de l'optimisation du seuil en fréquence et nous espérons obtenir de nouvelles améliorations en prenant en compte cet hyper-paramètre. Ces résultats restent inférieurs à ceux présentés dans (RUTHERFORD et XUE, 2015) qui utilisent une stratégie de sélection plus sophistiquée mais une simple union des données. Nous envisageons donc de mettre en œuvre une comparaison en testant également leur méthode de sélection combinée à nos stratégies de combinaison.

En multiclasse au niveau 1, nous obtenons au mieux un score de macro-F_1 de $43,8\%$ (exactitude de $51,8\%$) ce qui est supérieur au meilleur score de référence, des améliorations sont donc à attendre

en optimisant également le filtre en fréquence. Ce score est également supérieur à celui du système présenté dans (RUTHERFORD et XUE, 2015) qui correspond à une macro-F_1 de $40,5\%$ (exactitude de $57,1\%$). Comme nous n'optimisons pas le même score, la comparaison n'est pas directe mais nous avons pu observer que notre meilleur système correspondait à une amélioration pour toutes les relations sauf *Expansion*. Enfin, pour le niveau 2, notre meilleur système correspond à un score de macro-F_1 de $24,1\%$ (exactitude de $36,7\%$) inférieur au meilleur système de référence (macro-F_1 de $26,7\%$). Pour ce niveau, il nous semble qu'une étude additionnelle de la représentation doit être menée.

5 Utilisation de représentations denses pour l'identification des relations implicites

Sommaire

Nous avons présenté dans le chapitre précédent des expériences fondées sur l'ajout de données annotées automatiquement à partir des exemples explicites. Ces expériences se fondent sur des stratégies inspirées du cadre de l'adaptation de domaine afin de gérer les différences en termes distributionnels entre les données artificielles et naturelles. Cependant, l'apprentissage avec des données non identiquement distribuées est difficile. De plus, ces méthodes ont nécessité la définition d'heuristiques ou la construction de modèles afin d'extraire automatiquement les données supplémentaires. Dans ce chapitre, nous cherchons à améliorer les performances sans utiliser ces données artificielles tout en gardant le principe de se fonder sur une forme de non supervision. De plus, nous cherchons à évaluer la possibilité de construire un modèle reposant essentiellement sur une représentation surfacique des données limitant ainsi l'utilisation de ressources construites à la main.

La représentation surfacique correspond aux mots présents dans les arguments. Nous avons vu que l'information lexicale était un indicateur important de l'inférence des relations. Cependant, cette représentation souffre d'un problème d'éparpillement qui nous a conduit, dans le chapitre précédent, à chercher à enrichir le modèle à partir de nouvelles données. Ce problème provient de la façon dont est représentée cette information c'est-à-dire sous la forme d'une représentation dite *one-hot*. Avec un encodage one-hot, chaque terme, ici les mots ou les paires de mots, est associé à une dimension dans le modèle. La taille du modèle est donc égale au nombre de termes dans les données d'entraînement, la taille du vocabulaire, généralement assez large. Une instance est alors représentée par un vecteur dont la taille est celle du vocabulaire et dans lequel seules les dimensions correspondant aux termes présents dans l'instance reçoivent une valeur différente de zéro. On associe donc à chaque instance un vecteur de très haute dimension dans lequel seules quelques dimensions sont non nulles. Cette représentation très éparpillée pose problème dans le cadre d'un système d'apprentissage automatique. En effet, l'éparpillement rend l'estimation des paramètres du modèle difficile, peut conduire au problème de sur-apprentissage et rend difficile toute généralisation.

Afin de gérer cette difficulté, nous présentons dans ce chapitre des stratégies visant à rendre la représentation plus dense. Lorsque nous avions présenté, dans le chapitre 3, les études existantes pour l'identification des relations implicites, nous avions décrit plusieurs études reposant sur cette idée. Li et Nenkova (2014b) avaient proposé de réduire l'éparpillement pour les traits de type règles de production, la stratégie était alors fondée sur une ré-écriture du motif des traits. Une autre méthode possible est d'apprendre une transformation de la représentation plus dense liée à la tâche comme c'est notamment le cas dans (Ji et Eisenstein, 2014a). Cette stratégie est attrayante car elle lie la représentation au problème. Cependant, Ji et Eisenstein (2014a) ne parviennent pas à dépasser les scores rapportés dans (Rutherford et Xue, 2014) malgré la mise en place d'un système bien plus complexe et coûteux. La méthode de Rutherford et Xue (2014) est en effet assez simple puisqu'elle correspond comme dans (Li et Nenkova, 2014b) à une transformation du motif des traits. De plus, elle se fonde sur des représentations existantes pour les mots, elle ne nécessite donc pas une phase supplémentaire d'apprentissage, même si un apprentissage a dû être effectué pour construire cette représentation. Enfin, cette stratégie permet d'aboutir comme dans (Ji et Eisenstein, 2014a) à une représentation plus dense qui introduit une dimension sémantique et syntaxique dans la modélisation apportée par la représentation de mots utilisée. Cependant, Rutherford et Xue (2014) utilisent une représentation de mots clusterisée ce qui, nous allons le voir, permet de réduire le nombre de dimensions du modèle sans pour autant offrir une représentation dense à valeur réelle. Une telle représentation peut être obtenue en se fondant sur d'autres types de représentation des mots, distributionnelle ou distribuée. Nous explorons dans ce chapitre les effets de l'utilisation de ces différents types de représentation pour l'identification des relations implicites. Nous comparons également différentes stratégies de combinaison de ces représentations permettant d'obtenir une représentation pour des paires de segments textuels et l'utilisation de tous les mots ou seulement de certains mots des arguments, considérés comme particulièrement importants. Notons que l'un des attraits de la stratégie reposant sur les représentations de mots non supervisées repose sur le fait que ces représentations, qui sont disponibles librement pour l'anglais, peuvent être induites de manière non supervisée à partir de données brutes. Elles sont donc utilisables également pour les langues disposant de peu de ressources construites manuellement comme les lexiques de polarité ou de sentiment. Nous cherchons donc également à établir la nécessité de l'ajout d'autres traits à la représentation obtenue. Les résultats sur le PDTB en binaire au niveau 1 de sens ont été publiés dans (Braud et Denis, 2015).

Dans la section suivante 5.1, nous revenons sur le problème général de la représentation des données, de l'encodage one-hot et des difficultés liées. Nous présentons ensuite dans la section 5.2 les différentes représentations de mots existantes. La difficulté posée par notre tâche dans ce cadre est d'utiliser une représentation en mots pour représenter des instances composées de paires d'ensembles de mots. Dans la section 5.3, nous détaillons les difficultés posées par cette configuration et présentons des solutions fondées sur des opérations entre les vecteurs. Les expériences menées sur l'anglais sont décrites dans la section 5.4 : nous montrons en particulier que l'utilisation de ces représentations permet d'obtenir au niveau 1 de sens des performances proches de l'état de l'art voire meilleures sans utiliser d'informations issues de ressources construites à la main. Enfin, dans la section 5.6, nous proposons de construire une représentation de mots de type distributionnel en se fondant sur les connecteurs, donc liée à la tâche, et nous rapportons les performances obtenues en utilisant cette nouvelle représentation.

5.1 Problème de la représentation des données

Nous avons déjà discuté dans les précédents chapitres du problème d'éparpillement des données et des limites des stratégies reposant sur un usage intensif de ressources construites à la main. Plutôt que de considérer le problème sous l'angle d'un ajout de données afin d'obtenir une meilleure estimation des paramètres, nous considérons ici la possibilité de transformer une représentation simple sujette à l'éparpillement vers un espace plus dense. Plus précisément, nous nous intéressons à la représentation fondée sur les mots dans les arguments comme les paires de mots introduites par MARCU et ECHIHABI (2002) et généralement reprises par la suite dans les études sur le PDTB (PITLER et al., 2009 ; LIN et al., 2009 ; PARK et CARDIE, 2012 ; WANG et al., 2012 ; RUTHERFORD et XUE, 2014 ; RUTHERFORD et XUE, 2015). Ces traits ont été introduits pour identifier des paires de lexèmes pouvant déclencher une relation. Ainsi, dans l'exemple (55), issu du PDTB, la paire « *rose, tumbled* » (que l'on peut traduire par « *monter, chuter*«) signale une relation contrastive.

(55) [Quarterly revenue **rose** 4.5%, to $2.3 billion from $2.2 billion]$_1$ [For the year, net income **tumbled** 61% to $86 million, or $1.55 a share]$_2$

Souvent, c'est le fait qu'ils constituent une paire qui est le déclencheur : par exemple, la seule indication de la présence de « *pousser* » dans l'un des arguments de l'exemple (56) ne suffit bien sûr pas à inférer un lien causal, c'est parce que ce mot apparaît dans un argument et que l'autre contient « *tomber* » que l'on va inférer, éventuellement, un tel lien. De plus, si « *pousser* » est dans le premier argument et « *tomber* » dans le second, on identifiera plutôt une relation de type *Result* tandis que dans le cas inverse, on aura une relation de type *Explanation*, ces relations étant asymétriques.

(56) [Paul est tombé,] [Marie l'a poussé.]

Le fait d'effectuer un produit cartésien sur les mots des arguments, plutôt que, par exemple, de considérer les unigrammes sur l'ensemble des deux arguments, permet de conserver la relation d'ordre entre des évènements et de rapprocher des lexèmes qui, ensemble, fournissent un indice sur la relation. Notons que PARK et CARDIE (2012) concluent que ces traits ne sont plus utiles puisque des résultats au moins équivalents peuvent être obtenus en utilisant des représentations plus motivées, cependant linguistiquement fondées sur des ressources acquises manuellement et nécessitant d'importants pré-traitements.

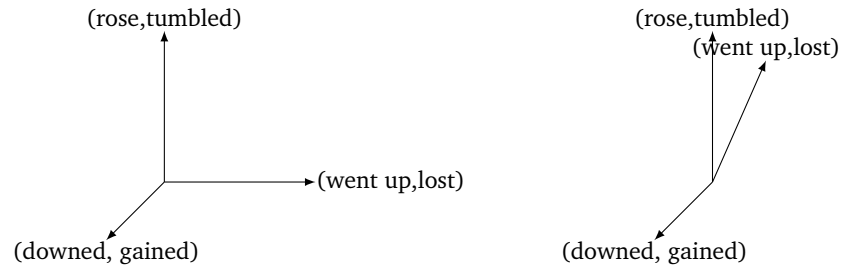


Figure 5.1.: Illustration d'une représentation des paires de mots : à gauche, avec un encodage one-hot tous les vecteurs d'instance sont perpendiculaires, alors que l'on voudrait, comme à gauche, que les paires de synonymes « *rose, tumbled* » et « *went up, lost* » soient plus proches l'une de l'autre que de la paire d'antonymes « *downed, gained* ».

Ces traits de paires sont toujours utilisés avec un encodage one-hot, éventuellement rendu plus dense par l'utilisation d'une représentation clusterisée (RUTHERFORD et XUE, 2014 ; RUTHERFORD et XUE, 2015). Si l'on considère le cas général où les instances sont directement représentées par les paires de mots, utiliser un encodage one-hot signifie que les instances sont associées à un vecteur dont la taille correspond au nombre de paires de mots dans les données. Si on suppose que tous les mots présents dans les données peuvent apparaître dans les arguments des instances, on se retrouve donc avec un vocabulaire dont la taille correspond au carré du nombre de mots dans les données. Chaque instance correspond donc à un vecteur de très haute dimensionnalité dans lequel seules les dimensions correspondant aux paires de mots représentées dans l'exemple ont une valeur non nulle, 1 si l'on se place dans le cadre d'un trait binaire [1].

Ce type de représentation est dite éparpillée, c'est-à-dire qu'elle contient beaucoup de 0, beaucoup plus que de valeurs non nulles. Cet éparpillement rend le modèle sujet au sur-apprentissage : le nombre de données d'entraînement étant largement inférieur au nombre de dimensions, le modèle aura tendance à apprendre des règles trop spécifiques aux données d'entraînement et ne sera pas capable d'une bonne généralisation. Le problème de généralisation est plus large puisque cette représentation ne permet pas au modèle de tenir compte des paires qui n'ont pas été vues à l'entraînement et ne dit rien des similarités potentielles entre les paires de mots. Par exemple, si l'on considère pour simplifier que l'on représente une instance seulement par la paire constituée par les verbes principaux de chacun de ses arguments, le vecteur one-hot correspondant à la paire « *rose, tumbled* », vu à l'exemple (55), sera à la même distance (euclidienne ou cosinus) des vecteurs représentants des instances correspondant à des paires de synonymes « *went up, lost* » ou d'antonymes « *downed, gained* », puisque ces trois vecteurs seront orthogonaux, comme illustré dans la figure 5.1. On voudrait au contraire que la représentation tienne compte des similarités entre les termes.

5.2 Représentations de mots

Une représentation de mots associe un mot à un objet mathématique, généralement un vecteur dont chaque dimension peut avoir une interprétation syntaxique ou sémantique. Dans cette section, nous présentons les différents types de représentation de mots en reprenant la terminologie de TURIAN et al. (2010).

1. On peut aussi utiliser la fréquence de la paire de mots dans l'exemple.

5.2.1 Représentation one-hot

La façon la plus brutale mais la plus commune de représenter les mots consiste à leur associer un vecteur dans lequel chaque mot observé correspond à une dimension distincte. Plus formellement, on note $\mathcal{V}$ l'ensemble de tous les mots trouvés dans les données d'entraînement et w un mot particulier de $\mathcal{V}$. La représentation one-hot de w est un vecteur indicateur de dimension d, noté $\mathbb{1}_w$ tel que $d = |\mathcal{V}|$: tous les composants de ce vecteur sont à 0 à l'exception d'un composant à 1 correspondant à l'indice de ce mot dans $\mathcal{V}$. Cette représentation est donc éparpillée et conduit à un modèle de très haute dimensionnalité.

5.2.2 Représentation fondée sur un clustering

Une alternative à cette représentation très éparpillée consiste à apprendre une représentation de façon non supervisée. On peut par exemple utiliser un algorithme de clustering pour regrouper les mots en classes. C'est l'approche utilisée pour construire les clusters *Brown* induits à partir d'un algorithme de clustering hiérarchique avec la visée de maximiser l'information mutuelle des bigrammes (BROWN et al., 1992). Chaque mot est alors associé à un code binaire représentant le cluster auquel il appartient. Cette représentation conduit également à un encodage one-hot des données mais à partir d'un vocabulaire plus petit correspondant au nombre de clusters. Étant donné que l'algorithme est hiérarchique, on peut utiliser des classes de mots de granularité différente qui correspondent à des codes de tailles différentes [2]. Plus le nombre de clusters est élevé, plus les distinctions entre les mots sont fines avec cependant un éparpillement plus élevé. Les clusters *Brown* ont été utilisés pour différentes tâches de TAL comme la reconnaissance d'entités nommées, le chunking (TURIAN et al., 2010), l'analyse syntaxique (KOO et al., 2008) ou l'identification des relations discursives implicites (RUTHERFORD et XUE, 2014).

5.2.3 Représentation distribuée

Une approche pour induire des représentations de mots à partir de données brutes consiste à apprendre une représentation distribuée. Ce type de représentation associe à chaque mot un vecteur dense, de faible dimensionnalité et à valeurs réelles. On appelle plongement lexical (*word embedding*) une telle représentation. Les plongements lexicaux sont généralement appris en utilisant des modèles de type réseaux de neurones (BENGIO et al., 2003). Chaque dimension correspond à un trait latent du mot qui capture une information de type paradigmatique. Les plongements *Collobert et Weston* sont un exemple d'une telle représentation apprise à partir de réseaux de neurones (COLLOBERT et WESTON, 2008). Ils sont appris en minimisant une perte entre le n-gramme courant et un n-gramme corrompu c'est-à-dire, pour l'implémentation que nous utilisons, dont le dernier mot provient du même vocabulaire mais est différent du mot du n-gramme original. COLLOBERT et WESTON (2008) montrent que les plongements lexicaux qu'ils proposent permettent d'atteindre des performances état de l'art pour la tâche d'étiquetage en rôles sémantiques sans utilisation de traits syntaxiques supplémentaires. Les plongements dits *Hierarchical Log-Bilinear* sont un autre exemple de représentation distribuée (MNIH et HINTON, 2007). Ils ont été induits en utilisant un modèle neural linéaire et probabiliste rendu plus rapide par l'utilisation d'un principe hiérarchique. Les plongements sont obtenus en concaténant les plongements des $n-1$ mots d'un n-gramme et en apprenant le plongement du dernier mot. TURIAN et al. (2010) testent ces deux types de représentations distribuées pour les tâches de chunking et de reconnaissance d'entités nommées. CHEN et al. (2013) comparent également ces deux plongements lexicaux pour différentes tâches de

2. Les codes binaires associés aux mots dans les clusters *Brown* sont de taille différente et peuvent notamment être de taille inférieure à la taille de code recommandée pour l'utilisation de ces clusters, on ne peut donc pas utiliser chaque élément du code comme une dimension et obtenir ainsi une représentation qui ne serait pas de type one-hot.

TAL dont l'identification de la polarité, du genre et du nombre des mots ainsi que l'identification des relations de synonymies et d'antonymies, tâches sur lesquelles ces plongements permettent d'obtenir des résultats supérieurs au système de référence.

5.2.4 Représentation distributionnelle

La dernière approche est fondée sur l'hypothèse que les mots qui apparaissent dans les mêmes contextes ont tendance à avoir des significations similaires. La construction d'une représentation distributionnelle commence par le calcul des fréquences brutes de co-occurence entre chaque mot et les $|\mathcal{D}|$ mots servant de contexte, avec $\mathcal{D}$ généralement plus petit que le vocabulaire entier. Une transformation est ensuite appliquée à ces comptes bruts, comme les mesures TF-IDF ou la mesure d'information mutuelle par point (PMI). Comme $|\mathcal{D}|$ est généralement trop large pour constituer une représentation utilisable en pratique, on utilise ensuite un algorithme de réduction de dimensions, on aboutit ainsi à une représentation à p dimension avec $p \ll |\mathcal{D}|$. Comme pour les représentations distribuées, la représentation finale pour un mot correspond à un vecteur dense, de basse dimensionnalité et à valeurs réelles. On appelle également plongement de mots une telle représentation. Un exemple récent d'une telle approche a été proposé dans (Lebret et Collobert, 2014) sous le nom de *Hellinger PCA*. La représentation est construite en utilisant l'algorithme d'analyse en composantes principales (*Principal Component Analysis*, PCA) (Pearson, 1901) comme algorithme de réduction de dimensions se fondant ici sur la mesure de distance Hellinger pour minimiser l'erreur de reconstruction des données. Un attrait important de ces approches réside dans le fait que les représentations sont plus rapides à construire que celles fondées sur des réseaux de neurones, comme celles présentées précédemment, tout en permettant des performances similaires sur les tâches de reconnaissance d'entités nommées et de classification de sentiment (Lebret et Collobert, 2014).

5.3 Construire une représentation au-delà du mot

Nous nous intéressons maintenant au problème de la construction d'une représentation pour les instances de relations implicites à partir des représentations de mots présentées précédemment, c'est-à-dire la combinaison des vecteurs représentant chaque mot en un vecteur composite représentant l'instance.

Les représentations que nous utilisons pour les paires de segments discursifs diffèrent selon trois dimensions. D'abord, nous comparons l'utilisation d'un seul mot par segment, approximativement les deux verbes principaux, et l'utilisation de tous les mots composant les segments. Ensuite, nous comparons des représentations éparpillées et denses, nous opposons donc un encodage one-hot et une représentation de basse dimensionnalité. Comme nous l'avons évoqué, les clusters *Brown* constituent un cas un peu spécial puisqu'ils permettent de réduire le nombre de dimensions mais correspondent toujours à un encodage one-hot. Enfin, nous utilisons deux types de combinaison à partir des vecteurs représentant les segments pour construire une représentation pour une instance : la concaténation et le produit de Kronecker. Nous comparons donc plus de configurations que dans (Rutherford et Xue, 2014), ces auteurs ne considérant qu'une représentation utilisant tous les mots des arguments à partir des clusters *Brown*.

5.3.1 Notations

Nos instances de classification sont des paires de segments textuels, les deux arguments de la relation à prédire. On note $S_1 = \{w_{1_1}, \ldots, w_{1_n}\}$ l'ensemble des n mots qui constituent le premier

segment et $S_2 = \{w_{2_1}, \ldots, w_{2_m}\}$ les m mots qui composent le second segment. Nous considérons donc chaque segment comme un sac de mots. On note toujours $\mathcal{V}$ le vocabulaire de mots, c'est-à-dire l'ensemble de tous les mots qui apparaissent dans les segments discursifs. Nous pourrons éventuellement nous référer à un sous-ensemble spécifique de $\mathcal{V}$. On note $\mathrm{head}(\cdot)$ la fonction qui extrait la tête d'un argument, nous détaillerons cette extraction dans la section 5.4.1.1, et $\mathcal{V}_h \subseteq \mathcal{V}$ l'ensemble des mots têtes.

Comme notre objectif est de comparer différentes représentations pour les paires de segments discursifs, nous définissons une fonction de représentation générique Φ qui retourne la représentation à d dimensions, un vecteur à valeur réelle, d'une paire de segments :

$$\begin{aligned} \Phi : \ & \mathcal{V}^n \times \mathcal{V}^m \to \mathbb{R}^d \\ & (S_1, S_2) \mapsto \Phi(S_1, S_2) \end{aligned}$$

Le but de l'apprentissage en binaire est de construire une fonction de classification $f_{\boldsymbol{w}}(\cdot)$ paramétrée par un vecteur $\boldsymbol{w} \in \mathbb{R}^d$ qui associe $\Phi(S_1, S_2)$ à une classe $c \in \{-1, +1\}$.

Nous avons déjà posé $\mathbb{1}_w$, le vecteur à d dimensions correspondant à un encodage one-hot pour un mot $w \in \mathcal{V}$. Nous utilisons de plus les symboles $\oplus$ et $\otimes$ pour représenter la concaténation et le produit de Kronecker, respectivement. Le produit de Kronecker entre deux vecteurs est équivalent à l'application du produit extérieur (*outer product*) $\boldsymbol{u}\boldsymbol{v}^\top \in \mathbb{R}^{m \times n}$. On transforme ensuite la matrice de taille $m \times n$ obtenue en un vecteur de taille $mn \times 1$ en emboîtant ses colonnes à l'aide de l'opérateur $\mathrm{vec}(\cdot)$.

5.3.2 Représentations fondées sur les têtes des arguments

L'une des représentations les plus simples que l'on peut construire pour une paire de segments (S_1, S_2) consiste à ne prendre en compte que les têtes de ces segments : $h_1 = \mathrm{head}(S_1)$ et $h_2 = \mathrm{head}(S_2)$. Dans cette configuration simple, il faut encore répondre à deux questions : (i) quelle représentation utilise-t-on pour h_1 et h_2, et (ii) comment combine-t-on ces représentations. Rappelons que l'ordre dans lequel les mots apparaissent est important, on veut donc que les combinaisons conservent cette information.

5.3.2.1 Représentation one-hot

Le cas le plus simple est d'utiliser un encodage de type one-hot pour les deux mots têtes, donc les vecteurs $\mathbb{1}_{h_1}$ pour la tête de l'argument 1 et $\mathbb{1}_{h_2}$ pour la tête de l'argument 2. On combine ensuite ces vecteurs par concaténation ou en utilisant le produit de Kronecker. Nous définissons donc les deux fonctions de représentation suivantes :

$$\Phi_{h,\mathbb{1},\oplus}(S_1, S_2) = \mathbb{1}_{h_1} \oplus \mathbb{1}_{h_2}$$

$$\Phi_{h,\mathbb{1},\otimes}(S_1, S_2) = \mathrm{vec}(\mathbb{1}_{h_1} \otimes \mathbb{1}_{h_2})$$

Avec ces définitions, on a $\Phi_{h,\mathbb{1},\oplus}(S_1, S_2) \in \{0,1\}^{2|\mathcal{V}_h|}$ et $\Phi_{h,\mathbb{1},\otimes}(S_1, S_2) \in \{0,1\}^{|\mathcal{V}_h|^2}$. La seconde représentation correspond à assigner un composant de valeur 1 à chaque paire de mots dans $\mathcal{V}_h \times \mathcal{V}_h$. C'est la représentation la plus éparpillée que l'on peut définir à partir des mots têtes. Dans un certain sens, c'est aussi la plus expressive puisque l'on apprend un paramètre pour chaque paire de mots, on capture donc les interactions entre les mots sur les segments. Au contraire, $\Phi_{h,\mathbb{1},\oplus}(S_1, S_2)$ ne modélise pas explicitement ces interactions puisqu'elle traite chaque mot d'un

segment donné par une dimension distincte. De plus, le produit de Kronecker est une opération non commutative, contrairement par exemple à la multiplication par dimension ou à la somme, ce qui permet de différencier les cas selon l'ordre des arguments (GREFENSTETTE et SADRZADEH, 2011).

On obtient le même type de représentation en utilisant les clusters *Brown*, la seule différence est que le vocabulaire est plus petit puisqu'il correspond au nombre de clusters différents.

5.3.2.2 Représentation dense

Une autre manière de représenter les mots têtes consiste à utiliser les vecteurs de basse dimensionnalité et à valeur réelle fournis par des représentations distributionnelles ou distribuées. Pour simplifier la notation, on appelle *plongement* un tel vecteur. On note $\boldsymbol{M}$ une matrice réelle de dimension $n \times p$ où la i^e ligne correspond au plongement de dimension p du i^e mot de $\mathcal{V}_h$, avec $p \ll |\mathcal{V}_h|$. Pour l'instant, on suppose que $n = \mathcal{V}_h$ ce qui n'est pas tout à fait vrai, nous discuterons du cas de mots inconnus en section 5.4.1.1. Avec cette matrice, nous pouvons dériver le plongement pour les mots têtes h_1 et h_2 à partir de leur représentation one-hot en utilisant une multiplication matricielle, respectivement $\boldsymbol{M}^\top \mathbb{1}_{h_1}$ et $\boldsymbol{M}^\top \mathbb{1}_{h_2}$. En utilisant à nouveau la concaténation et le produit de Kronecker, on obtient deux nouvelles fonctions de représentation :

$$\Phi_{h,\boldsymbol{M},\oplus}(S_1, S_2) = \boldsymbol{M}^\top \mathbb{1}_{h_1} \oplus \boldsymbol{M}^\top \mathbb{1}_{h_2}$$

$$\Phi_{h,\boldsymbol{M},\otimes}(S_1, S_2) = \text{vec}(\boldsymbol{M}^\top \mathbb{1}_{h_1} \otimes \boldsymbol{M}^\top \mathbb{1}_{h_2})$$

Ces représentations correspondent à un espace de plus basse dimensionnalité, on a $\Phi_{h,\boldsymbol{M},\oplus}(S_1, S_2) \in \mathbb{R}^{2p}$ et $\Phi_{h,\boldsymbol{M},\otimes}(S_1, S_2) \in \mathbb{R}^{p^2}$.

5.3.3 Représentations fondées sur tous les mots des arguments

Les représentations présentées pour les paires de mots têtes peuvent être généralisées au cas où l'on conserve tous les mots dans chacun des segments. La difficulté supplémentaire réside dans la combinaison des vecteurs représentant les différents mots de chaque argument et dans la normalisation qui devient ici nécessaire. Nous choisissons une solution simple pour le premier problème en représentant chaque segment par la somme sur les vecteurs représentant chaque mot. De nombreuses autres formes de combinaison sont possibles pour construire une représentation vectorielle pour une phrase, notamment pour prendre en compte les liens syntaxiques entre les mots des arguments, leur ordre et leur importance (MITCHELL et LAPATA, 2010 ; BLACOE et LAPATA, 2012). Il est également possible de construire directement des représentations pour les phrases (LE et MIKOLOV, 2014).

5.3.3.1 Représentation one-hot

En suivant cette approche, nous pouvons donc construire les représentations one-hot pour les paires de segments de la façon suivante, avec n le nombre de mots dans le premier argument S_1 et m le nombre de mots dans le second S_2 :

$$\Phi_{all,\mathbb{1},\oplus}(S_1, S_2) = \sum_i^n \sum_j^m \mathbb{1}_{w_{1_i}} \oplus \mathbb{1}_{w_{2_j}}$$

$$\Phi_{all,\mathbb{1},\otimes}(S_1, S_2) = \sum_i^n \sum_j^m \text{vec}(\mathbb{1}_{w_{1_i}} \otimes \mathbb{1}_{w_{2_j}})$$

Lorsque ces représentations sont utilisées sans aucune forme de filtre, elles correspondent à un espace de très grande dimensionnalité. On a $\Phi_{all,\mathbb{1},\oplus}(S_1, S_2) \in \mathbb{Z}_{\geq 0}^{2|\mathcal{V}|}$ et $\Phi_{all,\mathbb{1},\otimes}(S_1, S_2) \in \mathbb{Z}_{\geq 0}^{|\mathcal{V}|^2}$. La fonction de représentation $\Phi_{all,\mathbb{1},\oplus}(S_1, S_2)$ correspond à la représentation traditionnelle des paires de segments utilisée dans les études sur l'identification des relations depuis (MARCU et ECHIHABI, 2002).

De même que pour le cas où l'on utilisait seulement la tête des arguments, on a les mêmes fonctions de représentation en utilisant les clusters *Brown*, la différence résidant dans le fait que l'on a un espace de moins grande dimensionnalité qui ne dépend pas ici du nombre de mots mais du nombre de clusters.

5.3.3.2 Représentation dense

On peut utiliser les mêmes opérations de composition aux représentations denses. On aboutit aux fonctions de représentations suivantes :

$$\Phi_{all,\boldsymbol{M},\oplus}(S_1, S_2) = \sum_i^n \sum_j^m \boldsymbol{M}^\top \mathbb{1}_{w_{1_i}} \oplus \boldsymbol{M}^\top \mathbb{1}_{w_{2_j}}$$

$$\Phi_{all,\boldsymbol{M},\otimes}(S_1, S_2) = \sum_i^n \sum_j^m \text{vec}(\boldsymbol{M}^\top \mathbb{1}_{w_{1_i}} \otimes \boldsymbol{M}^\top \mathbb{1}_{w_{2_j}})$$

Ces fonctions de représentation sont dans le même espace que celles définies pour les représentations sur les têtes, $\Phi_{h,\boldsymbol{M},\oplus}(S_1, S_2)$ et $\Phi_{h,\boldsymbol{M},\otimes}(S_1, S_2)$, puisque la somme ne modifie pas la dimensionnalité des vecteurs que l'on combine. On a donc $\Phi_{all,\boldsymbol{M},\oplus}(S_1, S_2) \in \mathbb{R}^{2p}$ et $\Phi_{all,\boldsymbol{M},\otimes}(S_1, S_2) \in \mathbb{R}^{p^2}$

5.3.3.3 Normalisation des vecteurs

Il existe différentes façons de normaliser les vecteurs construits pour les paires de segments discursifs. La normalisation est importante ici car les vecteurs composites non normalisés sont sensibles au nombre de mots dans les arguments. Le premier type de normalisation que l'on considère consiste à simplement convertir les représentations vectorielles en vecteurs unitaires, ce que l'on obtient en divisant chaque vecteur par sa norme.

Concernant la représentation par concaténation, l'inversion des opérations de concaténation et de somme permet également d'obtenir une forme de régularisation. En effet, si l'on considère, par exemple, la fonction de représentation pour le cas one-hot que nous avons présenté précédemment, on peut la décomposer de la manière suivante :

$$\Phi_{all,\mathbb{1},\oplus}(S_1, S_2) = \sum_i^n \sum_j^m \mathbb{1}_{w_{1_i}} \oplus \mathbb{1}_{w_{2_j}} = m \sum_i^n \mathbb{1}_{w_{1_i}} \oplus n \sum_j^m \mathbb{1}_{w_{2_j}}$$

On a ici une forme de pondération de chacune des sommes, représentant chacune un argument, par le nombre de mots présents dans l'autre argument. On peut donc utiliser la version non pondérée

suivante qui consiste en une normalisation sur le nombre de mots, version que l'on peut construire pour toutes les fonctions présentées utilisant la concaténation :

$$\Phi'_{all,\mathbb{1},\oplus}(S_1, S_2) = \sum_i^n \mathbb{1}_{w_{1_i}} \oplus \sum_j^m \mathbb{1}_{w_{2_j}}$$

On peut bien sûr combiner cette forme de normalisation à celle correspondant au vecteur unitaire.

5.4 Configuration des expériences

5.4.1 Données

Notre objectif est d'évaluer la pertinence de différentes représentations vectorielles pour des instances de relations discursives implicites. Plus spécifiquement, nous voulons savoir si les représentations denses permettent de meilleures performances que les représentations éparpillées, si certains mots sont d'une importance cruciale pour l'identification des relations, quelles ressources et quelles stratégies de combinaison sont les plus adaptées à la tâche et, finalement, si les traits utilisés traditionnellement sont encore utiles lorsque l'on utilise une représentation dense des mots.

Nous utilisons le corpus du PDTB dans la même configuration que pour les expériences présentées dans le chapitre précédent. Nous présentons ici les différents types de traits utilisés, qui sont d'abord des traits lexicaux, et nous détaillons les paramètres des représentations de mots utilisées ainsi que leur couverture.

5.4.1.1 Ensemble de traits

Les traits utilisés sont d'abord lexicaux et se divisent en deux versions selon que l'on utilise seulement la tête verbale des arguments ou tous les mots présents dans les arguments.

Extraction des têtes

Nous définissons une heuristique pour extraire les têtes des arguments à partir de l'analyse en constituant fournie par le PTB. Les têtes syntaxiques sont d'abord extraites en utilisant les règles de percolation des têtes de Collins [3]. Afin de récupérer une tête sémantique, nous appliquons un ensemble de règles qui correspondent à chercher le participe passé d'un auxiliaire, l'attribut adjectival ou nominal du verbe copule « *be* » et le verbe à l'infinitif complément de « *have to* ». Dans le cas où l'on ne trouve pas de proposition contenant un verbe, on cherche d'abord un syntagme nominal, puis un syntagme adjectival et, finalement, un syntagme prépositionnel. Dans le cas d'une coordination ou d'un argument correspondant à plusieurs phrases, nous utilisons ces règles sur la première tête coordonnée. Quand un argument ne correspond pas à un seul sous-arbre bien formé, que l'on a donc de multiples sous-arbres, on cherche la tête de la première clause indépendante ou, s'il n'y en a pas, du premier syntagme. Ainsi, nous extrayons les têtes soulignées en gras dans les exemples suivants.

(57) a. [Trinidad Corp. is **contesting** liability.] [It **claims** the Coast Guard failed to chart the rock and refuses to pay damages.]

3. `https://github.com/jkkummerfeld/nlp-util`

b. [But such skills were **alien** to Toni Johnson.] [Four years ago, she was **managing** a film-processing shop and was totally bored.]

c. "We've done a lot to improve (U.S.) results [and a lot more will be **done**]," Mr. Mark said. "[Improving profitability of U.S. operations is an extremely high **priority** in the company.]"

d. [Lights **flickered** on and off ; plaster dropped from the ceiling, the walls still shook and an evacuation alarm blared outside.] [The four lawyers **climbed** out from under a table.]

e. The budget was only $400,000. "[**Athens** with Spartan means]," Mr. McDonough says. [The fund's lawyers **work** in an Athenian grove of potted trees.]

Cette extraction contient éventuellement des erreurs au sens où, par exemple, nous ne récupérons pas « *failed* » dans le second argument de l'exemple (58) mais le verbe introducteur de discours « *says* ». Dans cet exemple de la relation temporelle *Precedence*, on aurait plutôt voulu récupérer la tête du discours rapporté qui est à un temps passé contrairement au verbe du premier argument, différence qui peut signaler la relation d'asynchronie.

(58) [The screen **fills** with a small, tight facial shot of David Dinkins, Democratic candidate for mayor of New York City.] [David Dinkins failed to file his income taxes for four straight years," **says** a disembodied male voice.]

De plus, nous ne récupérons pas les auxiliaires, les modaux ou la présence d'une négation ou d'une particule (comme « *off* » dans « *set off* ») qui donnent pourtant des informations cruciales pour notre tâche. Pour l'instant, les informations temporelles, modales ou de négation sont encodées à l'aide de traits supplémentaires mais il faudra envisager dans de futures expériences de les inclure à la représentation dense [4]. De même, nous effectuons une simplification en ne conservant que la première tête en cas de coordination, il faudra envisager de combiner toutes les têtes.

Représentation fondée sur les mots

Nous utilisons soit un encodage one-hot, directement sur les mots (ou tokens), soit une représentation dense construite à partir des représentations distributionnelles ou distribuées. Les clusters *Brown* (*Brown*), les plongements Collobert-Weston (*CnW*) et hierarchical log-bilinear (*HLBL*) correspondent aux versions implémentées dans (TURIAN et al., 2010) [5]. Ces représentations ont été construites à partir du corpus journalistique anglais *Reuters*, la casse est laissée intacte. Nous testons les versions avec 100, 320, 1000 et 3 200 clusters pour *Brown*, avec 25, 50, 100 et 200 dimensions pour *CnW* et avec 50 et 100 dimensions pour *HLBL*. Les représentations Hellinger PCA (*H-PCA*) viennent de (LEBRET et COLLOBERT, 2014) [6]. Elles ont été construites à partir de *Wikipédia*, du corpus *Reuters* et du *Wall Street Journal*, tous les mots sont mis en minuscules. Le vocabulaire des *H-PCA* correspond aux mots qui apparaissent au moins 100 fois. La fréquence normalisée est calculée à partir des 10 000 mots les plus fréquents comme contexte. Nous testons les versions avec 50, 100 and 200 dimensions pour *H-PCA*. La couverture de chacune de ces ressources est présentée dans la table 5.1.

Taille des vocabulaires

Pour l'encodage one-hot, nous laissons la casse inchangée. La représentation induite à partir des clusters *Brown* est construite en ignorant les mots inconnus suivant (RUTHERFORD et XUE, 2014). Pour les plongements, on associe aux mots inconnus la moyenne des vecteurs sur tous les mots.

4. Nous ne pouvons effectuer une simple concaténation puisqu'une tête peut être associée à un nombre variable d'auxiliaires/modaux.

5. `http://metaoptimize.com/projects/wordreprs/`

6. `http://lebret.ch/words/`

	Nombre de mots	Nombre de mots manquants	
		Tous les mots	Mots têtes
HLBL	246 122	5 439	171
CnW	268 810	5 638	171
Brown	247 339	5 413	171
H-PCA	178 080	7 042	190

Table 5.1.: Couverture des représentations de mots utilisées : clusters *Brown* (*Brown*), plongement Collobert et Weston (*CnW*), plongement Hierarchical log-bilinear (*HLBL*), plongement Hellinger PCA (*H-PCA*).

Pour l'encodage one-hot, nous avons $|\mathcal{V}| = 33\ 649$ tokens différents dans les données[7], ce qui donne une idée de l'éparpillement de cette représentation. Les clusters *Brown* permettent d'opérer des regroupements de ces tokens qui correspondent alors à $3\ 190$ codes différents pour la version contenant originellement $3\ 200$ clusters, à 393 codes pour celle en contenant $1\ 000$, à 59 codes pour celle en contenant 320, ou à 16 codes pour celle en contenant 100.

Quand nous nous limitons aux têtes des arguments, nous comptons $5\ 615$ tokens différents qui correspondent à $1\ 988$ codes pour la version avec $3\ 200$ clusters et des nombres similaires aux précédents pour les autres versions.

Pour les représentations denses, la taille du vocabulaire correspond à deux fois le nombre de dimensions du plongement utilisé, soit entre 50 et 400, ou le carré de ce nombre, donc entre 625 et $40\ 000$.

Autres traits

Nous testons l'ajout de traits supplémentaires traditionnellement utilisés dans les études existantes sur la tâche. Ces traits correspondent aux catégories suivantes, telle que présentées dans le chapitre 3 : **Règles de production**, **Verbe**, **Modalité**, **Polarité**, **Catégories sémantiques** (Inquirer), **Nombre, pourcentage, dollars** et **Premier, dernier, trois premiers mots**. Tous ces traits sont représentés par un encodage one-hot, à l'exception de la longueur des syntagmes verbaux qui est un trait continu, nous les concaténons donc aux traits lexicaux. Cet ensemble correspond aux traits utilisés dans (RUTHERFORD et XUE, 2014), étude à laquelle nous voulons nous comparer.

5.4.2 Modèles

Nous utilisons le même algorithme de classification que dans le chapitre précédent, donc un algorithme linéaire par maximum d'entropie. Pour le niveau 1 de relation, on construit un classifieur binaire par relation. Nous gérons le problème de déséquilibre des classes en utilisant la stratégie de pondération des instances : chaque instance reçoit un poids inversement proportionnel à la fréquence de la classe à laquelle il appartient dans l'ensemble d'entraînement. Nous optimisons les mêmes hyper-paramètres que ceux décrits dans les expériences de référence en section 3.4.3 en regard de la F_1 dans le cas binaire et de ma macro-F_1 dans le cas multiclasse. Nous optimisons ici également le filtre en fréquence sur les traits. Notons que ce paramètre additionnel n'a de sens que pour les représentations non denses. La significativité statistique est testée en utilisant le t-test apparié et le test de Wilcoxon sur l'ensemble de test divisé en 20 sous-ensembles.

7. LI et NENKOVA (2014b) donnent des chiffres différents mais qui correspondent aux traits apparaissant plus de 5 fois dans les données.

5.5 Résultats

5.5.1 Expériences en binaire au niveau 1

Nous présentons dans cette section les résultats obtenus sur le PDTB pour des modèles en binaire au niveau 1 de sens. Nous rapportons les scores obtenus pour des modèles multiclasses dans les sections suivantes.

5.5.1.1 Utilisation de tous les mots

Dans cette configuration, nous avons trouvé que l'utilisation de représentations de mots acquises de manière non supervisée permet quasiment systématiquement d'obtenir des performances supérieures à celles obtenues en utilisant les mots bruts. Les scores obtenus sont présentés dans le tableau 5.2. Bien que la meilleure représentation diffère d'une relation à l'autre, le meilleur score de F_1 est toujours obtenu avec une représentation plus dense. Nos systèmes de référence correspondent à l'encodage one-hot directement sur les mots en utilisant les schémas de combinaison par concaténation ou par multiplication, notés *One-hot* $\oplus$ et *One-hot* $\otimes$, le dernier correspondant à la représentation utilisée la plus fréquemment dans les études existantes. Ces systèmes constituent une référence assez forte au sens où ils ont été obtenus en optimisant un filtre en fréquence, ce qui gère de manière brutale et simple l'éparpillement. Nos meilleurs systèmes fondés sur des représentations denses correspondent à des améliorations significatives en termes de F_1 d'environ 8% pour *Expansion*, 7% pour *Temporal* et $3,5\%$ pour *Contingency*. Les gains pour *Comparison* ne sont pas significatifs. Ces scores ont été obtenus en utilisant la normalisation au vecteur unitaire, et, pour les systèmes utilisant la concaténation, la normalisation par le nombre de mots dans les arguments, normalisations décrites dans la section 5.3.3.3. Cette forme de normalisation a permis d'obtenir les meilleures performances sur l'ensemble de développement.

Représentation	*Temporal*		*Contingency*		*Comparison*		*Expansion*	
	P	F_1	P	F_1	P	F_1	P	F_1
One-hot $\otimes$	$23,6$	$21,1$	$41,4$	$50,4$	$26,3$	$34,8$	$62,5$	$59,4$
One-hot $\oplus$	$17,9$	$23,0$	$41,4$	$51,3$	$25,0$	$34,1$	$62,3$	$59,0$
Brown $3,200$ $\otimes$	$18,0$	$20,4$	$40,8$	$50,9$	$27,6$	$34,8$	$62,4$	$61,2$
Best *Brown* $\otimes$	$18,7$	$15,5$	$43,4$	$\mathbf{53,8}^{**}$	$22,5$	$30,9$	$55,7$	$61,9$
Best *Brown* $\oplus$	$22,0$	$\mathbf{28,0}^{**}$	$38,5$	$49,5$	$21,8$	$31,2$	$53,4$	$\mathbf{67,4}^{**}$
Best *Embed.* $\otimes$	$17,0$	$23,0$	$42,6$	$52,8^{**}$	$24,6$	$\mathbf{35,0}$	$64,4$	$61,9$
Best *Embed.* $\oplus$	$15,6$	$26,0^{*}$	$42,6$	$52,5$	$22,7$	$33,1$	$62,8$	$60,2$

Table 5.2.: Modèles utilisant tous les mots sur les arguments sur l'anglais en binaire au niveau 1, précision (« P ») et F_1 par relation * $p \leq 0.1$, ** $p \leq 0.05$ comparé à *One-hot* $\otimes$ avec le t-test et Wilcoxon.

5.5.1.2 Comparaison des représentations denses

Nous avons obtenu les meilleures performances en utilisant la représentation clusterisée fondée sur les clusters *Brown* (systèmes *Brown*). Cela montre que cette ressource permet d'opérer des groupements sur les mots qui sont pertinents pour notre tâche. Rutherford et Xue (2014) décrivent de manière extensive l'intérêt de cette ressource pour notre tâche, avec, par exemple, le rapprochement de paires de mots correspondant au même cluster pour la relation *Expansion* ou la mise en lien de chiffres ou de dates pour les relations *Comparison* et *Temporal*. Cependant, la configuration reposant sur cette représentation utilisée dans (Rutherford et Xue, 2014) (*Brown* $3,200$ $\otimes$) ne conduit pas à des performances supérieures aux systèmes de référence utilisant les mots bruts, sauf pour *Expansion*. Rappelons que cette comparaison n'avait pas été mise en œuvre dans cette étude. Ceci nous conduit à penser que les améliorations rapportées dans cette étude ne proviennent pas

de l'utilisation des clusters. Quant aux plongements lexicaux (*Embed.*), bien qu'ils conduisent à des performances légèrement inférieures, ils permettent quand même des améliorations significatives pour *Temporal* et *Contingency*, et de légères améliorations pour les autres relations. Ceci montre que, même s'ils n'ont pas été construits en se fondant sur des critères discursifs, les dimensions latentes encodent des propriétés des mots qui sont pertinentes par rapport à leurs fonctions rhétoriques. La supériorité des clusters *Brown* par rapport aux plongements lexicaux rejoint les conclusions de TURIAN et al. (2010) pour deux autres tâches de TAL, la reconnaissance d'entités nommées et le chunking syntaxique.

De plus, TURIAN et al. (2010) ont montré que le meilleur plongement lexical était dépendant de la tâche. Nos expériences suggèrent que ce paramètre est dépendant de la relation : les meilleurs scores sont obtenus avec *HLBL* pour *Temporal*, avec *CnW* pour *Contingency*, avec *H-PCA* pour *Expansion* et avec *CnW* (meilleur système utilisant la multiplication) et *H-PCA* (meilleur système avec concaténation) pour *Comparison*. Ceci montre que ces quatre relations peuvent être considérées comme quatre tâches distinctes. L'identification de liens temporels ou causaux correspond à des indices très différents, les premiers reposant plutôt sur des expressions temporelles ou l'ordonnancement temporel des évènements tandis que les seconds reposent sur des informations lexicales ou des connaissances encyclopédiques sur les évènements. Nous pensons que ceci explique aussi que le comportement de la F_1 par rapport au nombre de clusters optimal pour *Expansion* soit vraiment différent de celui observé pour les autres relations : pour *Expansion*, le meilleur score est obtenu avec 100 clusters pour le système utilisant la concaténation et 320 pour le système basé sur une multiplication alors que pour les autres relations, les meilleures performances correspondent à l'utilisation de $1\,000$ ou $3\,200$ clusters. La relation *Expansion* est la moins sémantiquement marquée et elle profite donc moins de groupements sémantiques fins. Les figures 5.2 et 5.3 montrent ce comportement pour les systèmes utilisant respectivement la concaténation et la multiplication. On peut voir que les courbes pour les clusters *Brown* sont très différentes pour *Expansion* par rapport aux autres relations.

5.5.1.3 Comparaison des schémas de combinaison

La comparaison des schémas de combinaison montre d'abord que l'utilisation de l'encodage one-hot sur les mots bruts à partir d'une concaténation (*One-hot* $\oplus$), bien qu'elle n'encode pas les corrélations entre les mots, conduit à des performances similaires voire supérieures à celles obtenues en utilisant la forme multiplicative (*One-hot* $\otimes$) traditionnellement utilisée. Avec les clusters *Brown*, la forme concaténée permet de meilleurs scores de F_1 que la forme multiplicative sauf pour *Contingency* qui semble profiter de la modélisation des interactions.

En comparant les performances sur l'ensemble de développement, nous avons trouvé que les différences entre les deux formes de combinaison pour les clusters *Brown*, en excluant *Expansion*, dépendent du nombre de clusters utilisés. TURIAN et al. (2010) avaient trouvé que les performances augmentaient avec le nombre de clusters, les meilleurs scores étant obtenus avec $3\,200$ clusters. C'est aussi le cas pour notre tâche quand on utilise la concaténation comme on peut le voir sur les courbes en 5.2. Par contre, lorsque les vecteurs sont combinés par multiplication, la F_1 croît jusqu'à $1\,000$ clusters puis décroît comme le montrent les courbes en 5.3. Il y a bien sûr un compromis entre expressivité et éparpillement : utiliser trop peu de clusters conduit à des performances basses puisque l'on perd des distinctions importantes, mais en avoir trop conduit à une perte de généralisation. Pour les plongements lexicaux, les comportements sont plus consistants avec en général une amélioration des performances avec l'augmentation du nombre de dimensions prises en compte.

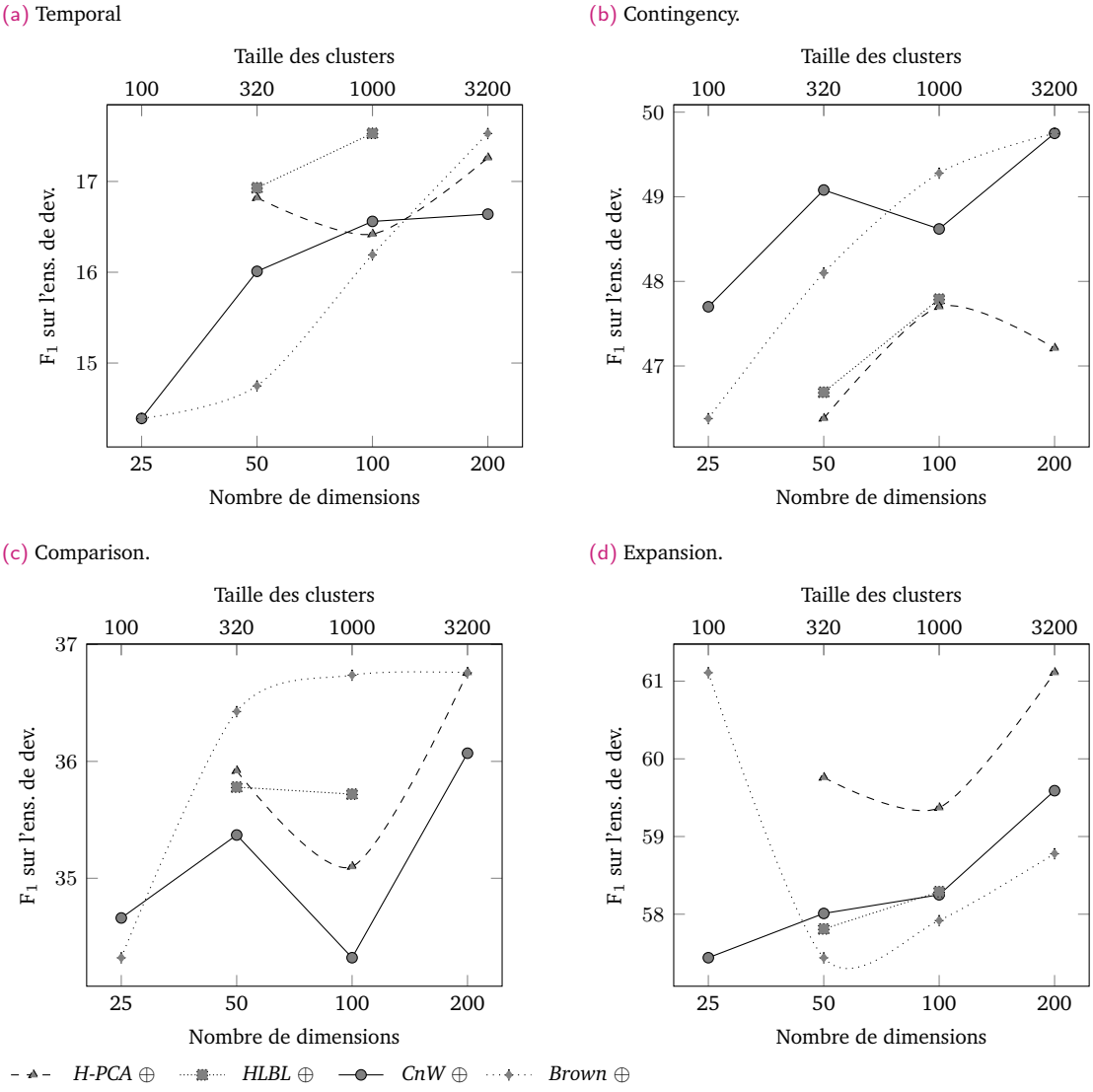


Figure 5.2.: Scores de F_1 sur l'ensemble de développement pour des systèmes en binaire au niveau 1 par rapport à la taille des clusters *Brown* et au nombre de dimensions des plongements de mots pour les systèmes utilisant la concaténation (systèmes $\oplus$).

5.5.1.4 Utilisation des têtes des arguments seules

Nous avons voulu tester si la restriction de la représentation à la seule tête sémantique des arguments suffisait à identifier une relation. En effet, dans certains cas, il semble que la seule paire des têtes soit un bon indicateur du lien comme dans l'exemple précédemment cité et repris en (59) où la paire « rose, tumbled » signale une relation contrastive.

(59) [Quarterly revenue **rose** 4.5%, to $2.3 billion from $2.2 billion]$_1$ [For the year, net income **tumbled** 61% to $86 million, or $1.55 a share]$_2$

Nous rapportons les résultats obtenus avec cette représentation dans la table 5.3. On observe que les performances sont inférieures à celles obtenues en utilisant tous les mots dans les arguments, et surtout pour le système de référence *One-hot* $\otimes$ qui correspond à des scores très bas avec une perte comprise entre 7% et 17% en termes de F_1. La baisse de performance est bien moins importante pour le système *One-hot* $\oplus$ et avec les représentations denses qui, de nouveau, correspondent

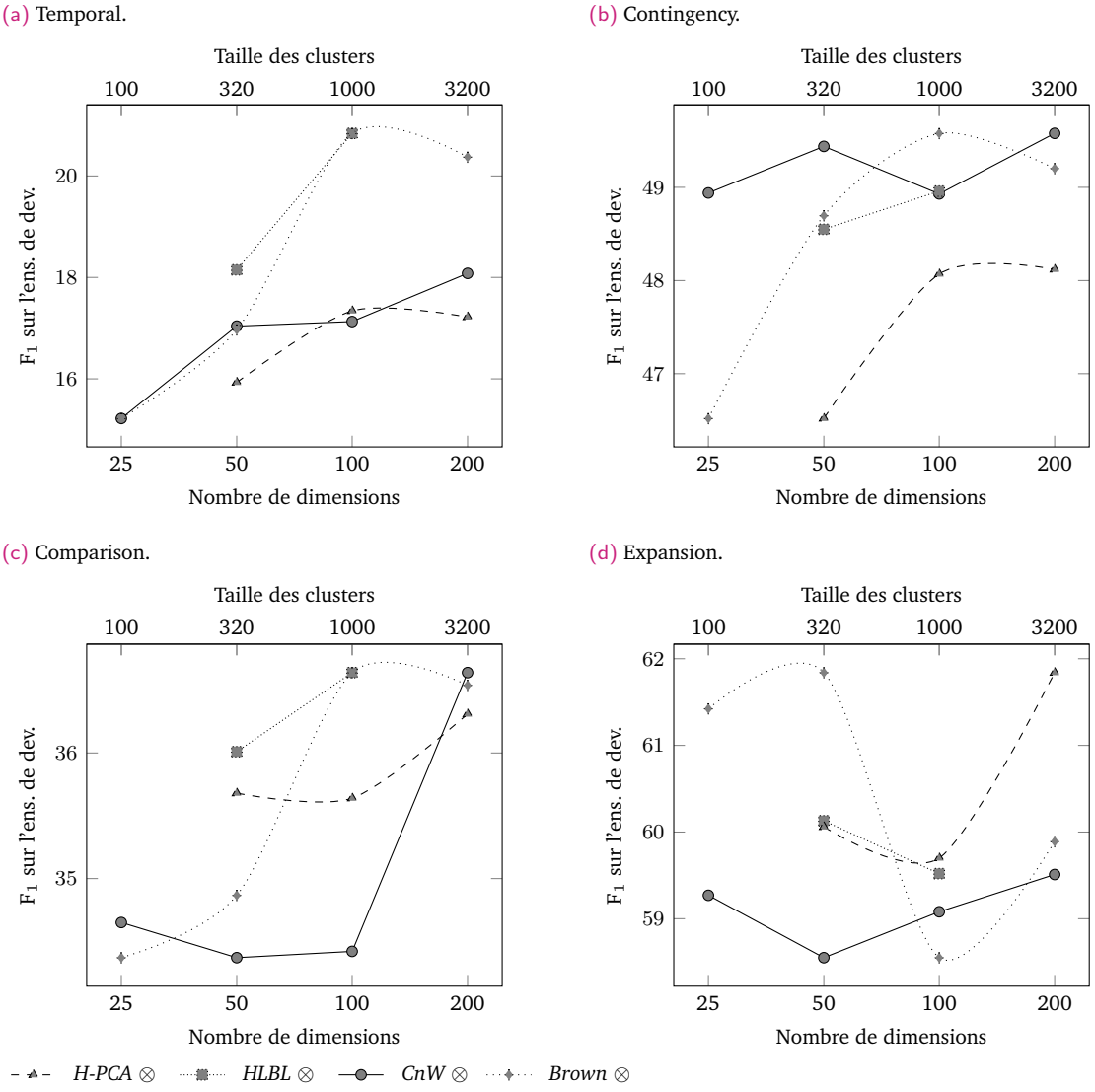


Figure 5.3.: Scores de F_1 sur l'ensemble de développement pour des systèmes en binaire au niveau 1 par rapport à la taille des clusters *Brown* et au nombre de dimensions des plongements de mots pour les systèmes utilisant le produit de Kronecker (systèmes $\otimes$).

aux meilleures performances. La seule exception est la relation *Expansion* dont la précision est cependant améliorée. Comme nous l'avons déjà dit, cette relation est la moins sémantiquement marquée, elle profite donc moins de l'utilisation de représentations de mots.

Les meilleures performances sont obtenues ici avec des plongements lexicaux et non avec les clusters *Brown*. On obtient des améliorations significatives comprises entre 8 et 13% en termes de F_1 pour la plupart des relations. De plus, les meilleurs systèmes sont tous fondés sur le schéma multiplicatif ce qui confirme que c'est un meilleur moyen de représenter les paires que la simple concaténation quand le nombre de dimensions initial n'est pas trop grand.

5.5.1.5 Ajout d'autres traits

Nous cherchons ici à savoir quelles améliorations nous pouvons obtenir en ajoutant aux traits construits à partir des représentations de mots d'autres traits traditionnellement utilisés. C'est également une façon d'évaluer les performances des systèmes fondés sur les représentations de mots uniquement en regard de l'état de l'art. Nous comparons nos résultats à ceux présentés

Représentation	*Temporal*		*Contingency*		*Comparison*		*Expansion*	
	P	F_1	P	F_1	P	F_1	P	F_1
One-hot $\otimes$	6,4	12,0	27,6	43,2	41,0	17,3	53,5	**69,2**
One-hot $\oplus$	16,5	23,0	35,4	49,4	22,1	29,2	57,1	59,1
Best *Brown* $\otimes$	16,3	22,9	31,6	45,7	15,0	25,8	53,4	68,8
Best *Brown* $\oplus$	14,2	21,8	34,7	47,4	17,9	27,5	54,5	61,4
Best *Embed* $\otimes$	18,0	**23,9**	39,8	**51,3**	20,3	**30,6**	56,8	58,6
Best *Embed* $\oplus$	13,3	22,5	36,9	47,5	19,7	29,8	56,6	57,4

Table 5.3.: Modèles utilisant seulement les têtes des arguments sur l'anglais en binaire au niveau 1, précision (« P ») et F_1 par relation. Toutes les améliorations observées par rapport au système *One-hot* $\otimes$ en termes de F_1 sont significatives.

dans (Rutherford et Xue, 2014) et dans (Ji et Eisenstein, 2014a), deux études cherchant à gérer le problème d'éparpillement soit en utilisant des clusters *Brown* soit en apprenant une représentation dense liée à la tâche. Pour rendre la comparaison plus facile, nous reproduisons les expériences de Rutherford et Xue (2014) avec l'algorithme naïf bayésien (NB) utilisé par ces auteurs [8] et avec l'algorithme par régression logistique (ME) que nous avons utilisé jusqu'à présent. Rappelons que le système proposé par ces auteurs correspond à l'utilisation des clusters *Brown* avec le schéma de combinaison par multiplication. Ces systèmes correspondent aux lignes « repr. » dans la table 5.4. La petite différence en termes de performance pour le système utilisant l'algorithme naïf bayésien doit être due au fait que nous n'incluons pas de traits de type coréférence et/ou à l'utilisation de seuils en fréquence différents. Concernant la différence entre les scores obtenus par les deux algorithmes, le seul vrai problème est la baisse en F_1 pour *Expansion* avec l'algorithme par régression logistique : en fait, le système utilisant l'algorithme naïf bayésien prédit tous les exemples comme positifs ce qui conduit à un score relativement haut tandis que le système construit avec l'algorithme par régression logistique produit des prédictions plus équilibrées. Aucun de ces deux systèmes n'est donc totalement satisfaisant. Finalement, nous donnons également des résultats en utilisant l'encodage one-hot traditionnel reposant sur les mots bruts et des traits supplémentaires (*One-hot* $\otimes$ + traits sup.). Tous ces résultats sont résumés dans la table 5.4, nous incluons également les scores rapportés dans la dernière étude sur le PDTB (Rutherford et Xue, 2015) et les meilleurs résultats obtenus précédemment, c'est-à-dire sans traits additionnels, pour faciliter la comparaison (lignes « seul. »).

	Temporal		*Contingency*		*Comparison*		*Expansion*	
System	P	F_1	P	F_1	P	F_1	P	F_1
(Rutherford et Xue, 2015)	-	**33,3**	-	53,8	-	41,0	-	69,4
(Ji et Eisenstein, 2014a)	-	26,9	-	51,4	-	35,8	-	**79,9**
(Rutherford et Xue, 2014)	18,5	28,7	44,5	54,4	27,3	**39,7**	59,6	70,2
repr. (Rutherford et Xue, 2014) NB	24,0	28,0	49,5	52,9	28,9	37,4	53,1	70,2
repr. (Rutherford et Xue, 2014) ME	28,3	24,8	44,4	53,4	30,0	36,5	53,2	50,0
One-hot $\otimes$ all tokens + traits sup.	24,6	23,3	46,3	54,4	30,6	34,3	64,9	62,6
Best all tokens seul.	22,0	28,0	43,4	53,8	24,6	35,0	53,4	67,4
Best heads seul.	18,0	23,9	39,8	51,3	20,3	30,6	53,3	69,2
Best all tokens + traits sup.	25,8	29,3	45,2	**55,8**	26,0	36,4	65,9	61,8
Best heads + traits sup.	23,8	22,9	43,2	54,1	26,2	36,5	65,3	61,8

Table 5.4.: Modèles utilisant des traits supplémentaires (« + traits sup. ») sur l'anglais en binaire au niveau 1, résultats état de l'art rapportés ou reproduits (« repr. ») en utilisant l'algorithme naïf bayésien (NB) ou par régression logistique (ME) et meilleurs systèmes des tableaux précédents (« seul. »), précision (« P ») et F_1 par relation.

8. Nous utilisons une autre implémentation, celle fournie dans le module scikit-learn.

Nous observons d'abord que l'ajout des traits supplémentaires permet d'obtenir des systèmes qui améliorent l'état de l'art pour *Temporal* et *Contingency*, et pour cette dernière correspondant aux meilleures performances actuelles. Ces améliorations sont significatives par rapport aux systèmes reproduits. Nous obtenons également de meilleures performances que celles rapportées dans (JI et EISENSTEIN, 2014a) utilisant un plongement lexical lié à la tâche sauf pour la relation *Expansion*. Une explication potentielle pour cette relation est que JI et EISENSTEIN (2014a) incluent les exemples EntRel et utilisent des traits de coréférence. Notons que nos systèmes correspondant à une reproduction de ceux présentés dans (RUTHERFORD et XUE, 2014) mènent à des résultats similaires aux systèmes de référence utilisant des paires de mots bruts (*One-hot* $\otimes$ all tokens+ traits sup.) montrant que les améliorations rapportées dans cette étude provenaient d'autres paramètres, comme l'optimisation d'un seuil en fréquence ou les traits de coréférence.

Ceci étant dit, l'ajout des traits supplémentaires à nos meilleurs systèmes utilisant tous les mots des arguments ne conduit pas à des améliorations aussi hautes que l'on pouvait s'y attendre. Bien que les améliorations soient significatives par rapport aux systèmes état de l'art reproduits, elles ne le sont pas par rapport à nos meilleurs systèmes sans ajout de traits supplémentaires (systèmes « seul. ») repris dans la table 5.4. Lorsque l'on utilise tous les mots des arguments, on a uniquement une tendance vers une amélioration significative pour *Contingency* ($p = 0,135$ avec le ttest et $p = 0,061$ avec le test de Wilcoxon). Ces différences très faibles montrent que les propriétés sémantiques et syntaxiques encodées dans ces traits sont déjà prises en compte dans les représentations de mots non supervisées que nous testons. Une étude supplémentaire sera nécessaire pour identifier quels traits exactement peuvent apporter des informations vraiment complémentaires à celles obtenues à partir de ces représentations.

Concernant l'utilisation des seules têtes des arguments, l'ajout des autres traits conduit à des améliorations importantes pour *Contingency* et *Comparison*, et aussi pour *Expansion* en termes de précision. Les résultats sont similaires pour *Temporal*. Notons que l'ajout des traits permet d'obtenir des résultats similaires aux systèmes utilisant tous les traits pour *Contingency*, *Comparison* et *Expansion* ce qui montre que tous les mots n'ont pas la même importance et que les têtes sémantiques sont d'une importance cruciale. Ceci renforce également l'idée que la seule prise en compte de ces têtes n'est pas suffisante, nous devons y intégrer d'autres types d'informations notamment d'ordre temporel et modal ici incluses dans les autres traits. Dans de futures expériences, il faudra construire un ensemble de mots autour des têtes et trouver une façon de représenter cet ensemble pouvant varier en taille de manière à avoir une similarité pour des groupements comportant un même modal ou correspondant à une même situation temporelle (c'est-à-dire distinguer « *have been* » et « *had been* » mais avoir une similarité entre « *was* » et « *had* »).

5.5.1.6 Courbes d'apprentissage

On peut se demander si l'utilisation de représentations denses permet de diminuer le nombre d'exemples d'entraînement nécessaire. Nous présentons dans la figure 5.4 les scores de F_1 pour chaque relation sur l'ensemble d'évaluation en faisant varier la taille des données d'entraînement entre 10 et 100% des données disponibles. Plus précisément, les scores pour les sous-ensembles inférieurs à 100% correspondent à une moyenne sur 10 expériences. La ligne horizontale pointillée correspond au score de référence obtenu avec l'ensemble des données pour la représentation *One-hot* $\otimes$.

On voit clairement pour *Temporal* et *Contingency* que l'apport de données d'entraînement conduit à des améliorations ce qui est attendu dans le sens où les meilleurs systèmes pour ces relations se fondent sur les clusters *Brown* donc une représentation qui demeure sujette à l'éparpillement. On note cependant qu'avec $70 - 80\%$ des données, on obtient déjà des performances assez hautes et

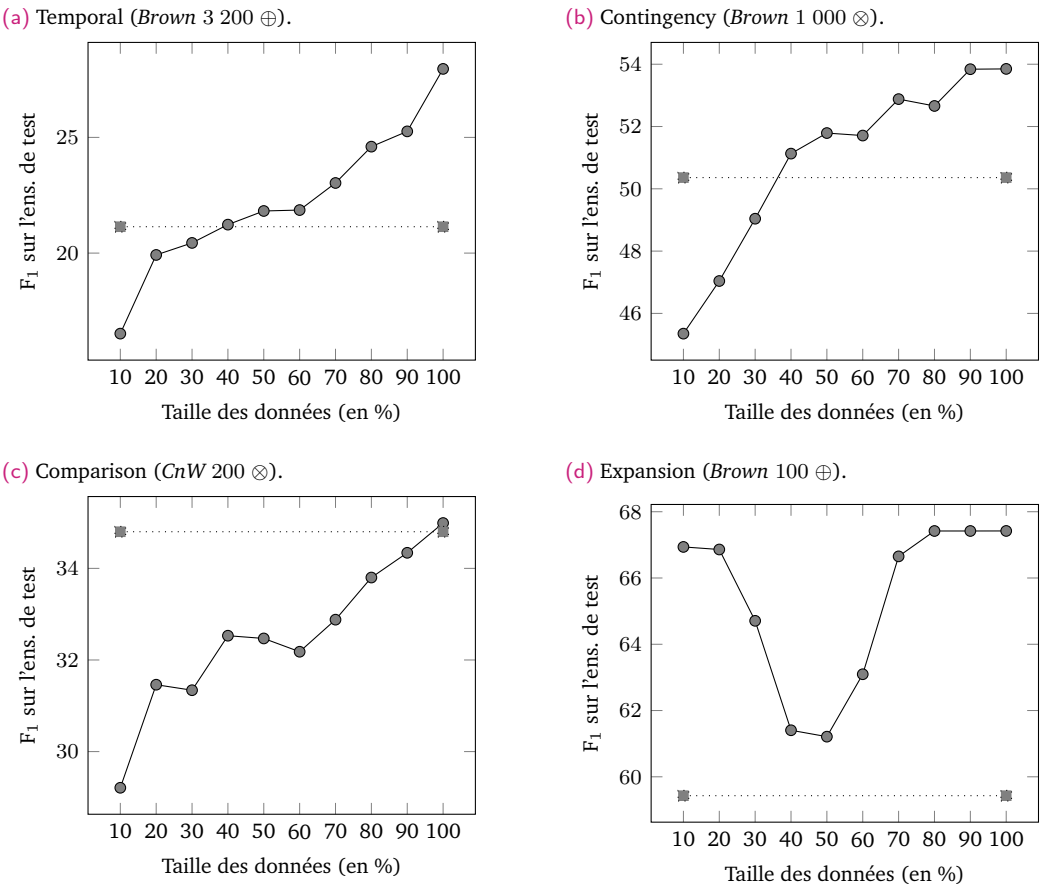


Figure 5.4.: Scores de F_1 des meilleurs systèmes obtenus sur l'ensemble d'évaluation en binaire au niveau 1 par rapport à la taille des données d'entraînement. Nous ajoutons en pointillés le score obtenu avec le système *One-hot* ⊗ sur l'ensemble des données.

que l'on dépasse le système de référence dès $40-50\%$ des données d'entraînement. Pour la relation *Comparison*, nous n'avons pas obtenu d'amélioration importante en utilisant une représentation plus dense. Même s'il semble que l'on puisse atteindre des résultats au niveau du système de référence en utilisant un sous-ensemble des données d'entraînement, le nombre de données nécessaire reste important (plus de 80%) malgré l'utilisation d'un plongement lexical dense. Pour la relation *Expansion*, la courbe correspond à un comportement moins prévisible, comportement qui suit celui du rappel. Il n'est cependant pas évident de comprendre pourquoi on observe une si large baisse du rappel en utilisant entre 30 et 60% des données. De manière générale, le comportement des performances pour cette relation est toujours différent des autres, ce qui vient en partie de sa faible caractérisation sémantique et de sa sur-représentation dans les données.

5.5.2 Expériences en multiclasse au niveau 1

Nous testons les stratégies présentées précédemment dans des systèmes multiclasses au niveau 1 de la hiérarchie de sens du PDTB, les résultats pour le niveau 2 seront présentés dans la section suivante. Dans ce cadre, nous optimisons les mêmes hyper-paramètres que précédemment mais en regard du score de F_1 macro-moyenné.

5.5.2.1 Utilisation de tous les mots des arguments

L'utilisation des représentations de mots fondées sur les clusters *Brown* ou sur les plongements lexicaux conduit à des scores similaires voire supérieurs à l'utilisation des paires de mots bruts

(*One-hot* $\otimes$). La légère baisse observée pour le meilleur système fondé sur les clusters *Brown* avec le schéma multiplicatif (Best *Brown* $\otimes$) n'est pas significative (t-test $p = 0,666$, Wilcoxon $p = 0,709$). Les meilleurs systèmes en termes de macro-F_1 sont obtenus avec les plongements lexicaux, et non les clusters *Brown* comme en binaire. Le meilleur système correspond à la combinaison par concaténation (Best *Embed.* $\oplus$) utilisant la représentation *CnW* 200, l'amélioration observée de $2,61\%$ en macro-F_1 correspond à une tendance vers une amélioration significative (t-test $p = 0,09$, Wilcoxon $p = 0,108$).

Dans la dernière étude sur le PDTB en multiclasse au niveau 1, RUTHERFORD et XUE (2015) rapportent $38,4\%$ de macro-F_1 et $55,0\%$ de micro-exactitude pour leur système de référence utilisant le clusters *Brown* et des traits additionnels. Leur meilleur système, utilisant des données artificielles, correspond à une macro-F_1 de $40,5\%$ et une micro-exactitude de $57,1\%$. Nous obtenons un score de macro-F_1 du même ordre avec notre système de référence (*One-hot* $\otimes$), donc en utilisant uniquement les traits de type paires de mots, et nous améliorons ce score en utilisant les plongements lexicaux. Notre micro-exactitude est cependant plus basse, rappelons que nous optimisons la macro-F_1 tandis que ces auteurs optimisent la micro-exactitude.

Représentation	Macro-prec	Macro-F_1	Exactitude
One-hot $\otimes$	38,6	39,0	48,6
One-hot $\oplus$	39,8	40,2	50,2
Best *Brown* $\otimes$	38,7	37,5	50,6
Best *Brown* $\oplus$	40,2	40,6	51,2
Best *Embed.* $\otimes$	41,4	41,0	51,7
Best *Embed.* $\oplus$	41,4	**41,6**	50,1
Best all tokens + traits sup.	40,7	40,8	51,2
Combine	42,3	40,6	53,4

Table 5.5.: Modèles utilisant tous les mots sur les arguments sur l'anglais en multiclasse au niveau 1, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »)), et scores d'exactitude de chaque système.

Concernant les scores par relation, notre meilleur système (Best *Embed.* $\oplus$) permet d'obtenir des améliorations pour la majorité des relations (voir tableau 5.6b) par rapport aux scores obtenus avec le système de référence (*One-hot* $\otimes$) pour lequel les résultats sont présentés dans le tableau 5.6a. On observe ainsi une augmentation du score de F_1 de $6,5\%$ pour *Temporal*, de $3,6\%$ pour *Contingency* et de $0,6\%$ pour *Expansion*. Comme dans le cas binaire, on n'obtient pas d'amélioration pour la relation *Comparison*. Nous rapportons également les meilleurs scores présentés dans (RUTHERFORD et XUE, 2015) dans le tableau 5.6c. Pour rappel, dans cette étude, les auteurs utilisent des données artificielles pour augmenter le nombre de données disponibles et une représentation similaire au jeu de traits **base+lex+synt** avec ajout de traits de coréférence et transformation des traits de type paires de mots en utilisant les clusters *Brown*. Nous obtenons des résultats supérieurs à ceux présentés dans cette étude pour *Temporal* et *Contingency* montrant qu'une bonne représentation peut finalement avoir un impact plus important qu'un ajout massif de nouvelles données. Notons que pour *Expansion* notre score de précision est meilleur, donc moins d'instances des autres relations sont prédites comme exemple d'*Expansion*, mais notre rappel est beaucoup plus bas, indiquant probablement une tendance du système de RUTHERFORD et XUE (2015) à prédire par erreur des exemples vers cette relation sur-représentée.

L'ajout des traits supplémentaires (Best all tokens + traits sup.) au meilleur système (Best *Embed.* $\oplus$) n'apporte ici aucune amélioration en terme de macro-F_1. On observe une amélioration de l'exactitude mais les scores par relation sont inférieurs à ceux obtenus avec le meilleur système sauf pour la relation *Expansion* (F_1 de $64,0\%$). Ces traits ne permettent donc pas une meilleure discrimination des classes ce qui rejoint nos conclusions en binaire : les traits traditionnellement

utilisés ne sont plus utiles lorsqu'on utilise une représentation de mots et un schéma de combinaison adaptés.

(a) *One-hot* $\otimes$.

Rel	P	R	F_1
Temp	15,6	25,0	19,2
Cont	44,0	41,3	42,6
Comp	32,1	34,9	33,4
Expa	62,8	58,6	60,6

(b) Best *Embed.* $\oplus$.

Rel	P	R	F_1
Temp	19,0	39,7	**25,7**
Cont	47,5	44,9	**46,2**
Comp	35,1	31,5	33,2
Expa	63,9	58,8	61,2

(c) (Rutherford et Xue, 2015).

Rel	P	R	F_1
Temp	38,5	9,1	14,7
Cont	49,3	39,6	43,9
Comp	44,9	27,6	**34,2**
Expa	61,4	78,8	**69,1**

Table 5.6.: Scores de précision (« P »), rappel (« R ») et F_1 par relation pour les systèmes sur l'anglais en multiclasse au niveau 1 : système de référence *One-hot* $\otimes$ et meilleur système Best *Embed.* $\oplus$ utilisant tous les mots sur les arguments. Nous rapportons également les scores du meilleur système présenté dans (Rutherford et Xue, 2015).

Comme nous avons vu dans les expériences en binaires précédemment présentées que le meilleur système (type de représentation et de combinaison) dépendait de la relation, nous testons également, pour le multiclasse, une combinaison des meilleurs systèmes construits en binaire. Plus précisément, pour chaque relation, nous disposons d'un vecteur par instance construit en se fondant sur les clusters *Brown* ou sur un plongement lexical. Nous concaténons le vecteur construit pour chaque instance pour un système binaire donné avec celui construit pour les autres systèmes binaires. En multiclasse au niveau 1 nous avons donc pour chaque instance quatre blocs dans la représentation correspondant chacun au meilleur système construit pour chacune des quatre classes. Les résultats pour cette combinaison sont repris dans la ligne « Combine » dans le tableau 5.5. Nous observons une légère amélioration en termes de macro-F_1 et une amélioration de presque 5 points en micro-exactitude par rapport au système de référence (*One-hot* $\otimes$) mais les scores sont cependant inférieurs à ceux obtenus avec notre meilleur système (Best *Embed.* $\otimes$).

Représentation	Macro-prec	Macro-F_1	Exactitude
One-hot $\otimes$	30,1	20,6	52,3
One-hot $\oplus$	36,5	36,0	47,6
Best *Brown* $\otimes$	32,0	30,4	46,5
Best *Brown* $\oplus$	36,1	35,6	47,8
Best *Embed.* $\otimes$	37,5	**36,8**	48,9
Best *Embed.* $\oplus$	36,9	36,6	47,2

Table 5.7.: Modèles utilisant seulement les têtes des arguments sur l'anglais en multiclasse au niveau 1, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système. Toutes les améliorations observées par rapport à *One-hot* $\otimes$ sont significatives.

5.5.2.2 Utilisation des têtes des arguments seules

Lorsque l'on se limite aux seules têtes des arguments (tableau 5.7), on observe comme en binaire une large baisse des performances avec le système de référence *One-hot* $\otimes$, on perd notamment environ 18 points de macro-F_1 par rapport au système utilisant tous les mots des arguments. Ici, ce système de référence correspond en fait à une prédiction de quasiment tous les exemples vers la relation majoritaire, *Expansion*, comme le montrent les scores par relations repris dans le tableau 5.8a. On observe également que le système de référence fondée sur la concaténation (*One-hot* $\oplus$) conduit à une amélioration très importante par rapport au système utilisant la forme multiplicative, avec un gain d'environ 10 points de macro-F_1et des prédictions plus équilibrées avec pour toutes les relations des scores de F_1 supérieurs à 20% (voir tableau 5.8b). Ici encore, ce sont les plongements lexicaux qui conduisent aux meilleures performances améliorant significativement les scores du système de référence *One-hot* $\otimes$ en termes de macro-F_1. On observe également

une importante amélioration de l'identification de la relation *Temporal* qui semble profiter tout particulièrement de l'utilisation des représentations de mots (voir tableau 5.8c).

(a) *One-hot* ⊗.

Rel	P	R	F_1
Temp	0,0	0,0	0,0
Cont	27,3	1,1	2,1
Comp	40,0	6,8	11,7
Expa	53,2	96,0	68,5

(b) *One-hot* ⊕.

Rel	P	R	F_1
Temp	17,5	25,0	20,6
Cont	47,8	35,1	**40,5**
Comp	22,7	21,9	**22,3**
Expa	58,2	63,3	60,6

(c) Best *Embed.* ⊗.

Rel	P	R	F_1
Temp	22,7	36,8	**28,1**
Cont	40,5	37,7	39,0
Comp	27,1	13,0	17,6
Expa	59,8	65,5	**62,5**

Table 5.8.: Scores de précision (« P »), rappel (« R ») et F_1 par relation pour les systèmes de référence sur l'anglais en multiclasse au niveau 1 : *One-hot* ⊗, *One-hot* ⊕ et meilleur système Best *Embed.* ⊗ utilisant seulement les têtes des arguments.

5.5.3 Expériences en multiclasse au niveau 2

Nous testons enfin les représentations denses pour des systèmes en multiclasse au niveau 2. Les scores obtenus sont rapportés dans le tableau 5.9. Ici, cependant, aucun système ne conduit à une amélioration de la macro-F_1 par rapport au système de référence *One-hot* ⊗, à l'exception d'une légère amélioration avec le système par concaténation *One-hot* ⊕. Comme le problème d'éparpillement est plus important en augmentant le nombre de classes, on aurait pu s'attendre au contraire à ce que les représentations denses soient particulièrement utiles ici. Ces résultats montrent donc que le problème doit venir de l'information apportée par les représentations : les distinctions plus fines correspondant au second niveau de sens ne se reflètent pas dans les représentations de mots utilisées ce qui montre la nécessité de les adapter à la tâche. Les systèmes limités aux têtes des arguments, dont les scores sont repris dans le tableau 5.10, obtiennent à nouveau des scores très inférieurs à ceux obtenus en utilisant tous les mots et, comme précédemment, la simple utilisation d'une forme concaténée, moins sujette à l'éparpillement permet d'améliorer largement les scores, tout comme en général l'utilisation de représentations plus denses.

Représentation	Macro-prec	Macro-F_1	Exactitude
One-hot ⊗	32,5	23,7	38,4
One-hot ⊕	23,4	**24,1**	35,5
Best *Brown* ⊗	23,1	20,1	34,9
Best *Brown* ⊕	21,7	22,2	33,5
Best *Embed.* ⊗	21,3	20,3	29,2
Best *Embed.* ⊕	21,5	20,5	28,2
Best all tokens + traits sup.	21,2	21,2	34,6

Table 5.9.: Modèles utilisant tous les mots sur les arguments sur l'anglais en multiclasse au niveau 2, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système.

Représentation	Macro-prec	Macro-F_1	Exactitude
One-hot ⊗	15,1	6,8	27,9
One-hot ⊕	16,3	**15,3**	24,8
Best *Brown* ⊗	20,3	7,1	24,4
Best *Brown* ⊕	15,5	13,8	22,2
Best *Embed.* ⊗	15,9	13,9	21,3
Best *Embed.* ⊕	16,5	14,4	21,3

Table 5.10.: Modèles utilisant seulement les têtes des arguments sur l'anglais en multiclasse au niveau 2, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système.

5.6 Plongement lexical à partir des connecteurs

Les représentations de mots utilisées dans les expériences précédentes ont été apprises ou construites de manière non supervisée et sans aucun lien avec la tâche d'identification des relations discursives. Nous avons vu que, malgré cela, les informations qu'elles encodent pouvaient être pertinentes pour l'identification des liens de type sémantico-pragmatique qui nous intéressent ici, du moins pour les grandes classes de relations. Il a cependant été montré, pour l'analyse de sentiment (LABUTOV et LIPSON, 2013) par exemple, que des représentations de mots plus adaptées à la tâche permettaient de meilleures performances. En effet, les plongements lexicaux que nous utilisons effectuent des rapprochements paradigmatiques non pertinents par rapport à notre tâche : par exemple, des adjectifs comme « *small* » et « *big* » seront rapprochés car fonctionnant dans des contextes similaires alors qu'ils constituent pour nous une paire de mots opposés dont la présence peut marquer un contraste.

Il existe différentes façons de construire des représentations adaptées à une tâche. Une possibilité est de lier l'apprentissage des représentations à celui de la tâche de classification en définissant un objectif joint comme l'ont fait MAAS et al. (2011) pour la tâche de classification de sentiments. La solution proposée par LABUTOV et LIPSON (2013) pour cette même tâche est plus simple et plus rapide puisqu'elle consiste à partir d'une représentation existante qui est modifiée en lien avec l'erreur de classification, cette transformation étant donc liée à la tâche. Nous évaluons dans cette section une autre possibilité consistant à construire une représentation de type distributionnel à partir des connecteurs discursifs qui servent d'indicateur pour les relations qu'ils déclenchent.

5.6.1 Principe

La construction d'une représentation distributionnelle se fonde sur une hypothèse dite distributionnelle stipulant que les mots qui apparaissent dans un même contexte ont tendance à avoir un sens similaire (HARRIS, 1954). Nous avons déjà dit que les connecteurs étaient de bons indices des relations. On peut en quelque sorte considérer les relations comme des clusters de connecteurs : dans environ 94% des cas, le connecteur suffit à identifier la relation. On peut donc penser que le connecteur constitue un contexte pertinent pour lier un mot à une relation discursive. Nous construisons une représentation distributionnelle en considérant les mots apparaissant dans les arguments d'une relation explicite dans le contexte du connecteur utilisé pour lexicaliser la relation, ce qui nous permet d'aboutir à une ressource du même type que celles utilisées précédemment : nous associons un mot à un vecteur dont les dimensions sont liées à leurs propriétés rhétoriques à travers les connecteurs. Cette stratégie ressemble à la méthode utilisée dans (BIRAN et MCKEOWN, 2013) mais ces auteurs n'utilisent pas directement la représentation construite comme un plongement lexical, ils utilisent cet espace pour calculer une mesure de similarité entre une représentation vectorielle d'un connecteur et une instance de relation implicite. Nous utilisons plutôt les méthodes de combinaison mises en place dans les précédentes expériences. Notons cependant que ces auteurs se fondent sur les paires de mots et non les mots, ce qu'il serait intéressant de tester dans de futures expériences. Cette méthode s'inspire également des travaux de CONRATH et al. (2014) qui consistent à construire une ressource associant à une paire de verbes des mesures d'association avec des relations de discours. Nous envisageons de tester cette ressource dans de futures expériences, notons cependant que nous ne pouvons l'utiliser comme seule source de traits puisqu'elle ne permet pas d'attribuer une représentation aux exemples dont l'un des arguments n'a pas une tête verbale. Notons également que la construction d'un plongement à partir de paires de mots nécessite, par rapport à l'utilisation de mots simples, de prendre en compte un plus grand nombre de données afin d'obtenir une bonne estimation de la corrélation entre paires et relation.

La construction d'une représentation distributionnelle se fait en plusieurs étapes (SAHLGREN, 2006 ; TURNEY et PANTEL, 2010). Dans un premier temps, on effectue un ou plusieurs pré-traitements sur le texte brut : d'abord, la tokenisation définit le type de terme utilisé, ensuite une forme de normalisation peut être utilisée pour supprimer par exemple la capitalisation, enfin on peut vouloir annoter le texte pour distinguer par exemple des formes ambigües (comme « *fly* » en tant que nom ou verbe), ce qui représente un effort important.

Dans un second temps, on construit une matrice de fréquence dans laquelle un élément correspond à un *évènement* : un certain objet (terme, mot, ...) apparaît dans une certaine situation (contexte, motif, ...) un certain nombre de fois (fréquence). Les éléments de cette matrice sont ensuite pondérés afin d'associer un poids plus important à des évènements surprenants et moins de poids à des évènements attendus. En effet, les évènements surprenants sont plus discriminants : la présence de « *the* » dans un argument a peu d'importance, ce mot devrait apparaître aussi souvent avec toutes les relations, par contre la construction de la ressource doit mettre en évidence des mots apparaissant plus fréquemment avec certains connecteurs qu'avec d'autres. La façon la plus commune d'effectuer ces pondérations consiste à utiliser la mesure dite TF-IDF (*Term Frequency-Inverse Document Frequency*, voir équation (5.5)). Avec cette mesure, un élément reçoit un poids important quand le terme correspondant est fréquent par rapport au document (ici le connecteur) donc si TF est grand, mais que le terme est rare dans les autres documents (donc pour les autres connecteurs) donc si IDF est bas. Une autre fonction de pondération utilisée est fondée sur la mesure d'information mutuelle ou PMI (*Pointwise Mutual Information*, voir équation 5.2). La mesure d'information mutuelle est une mesure de l'information que donne une variable aléatoire à propos de la valeur d'une autre variable aléatoire. Elle est souvent utilisée pour la sélection de traits. La mesure PMI associe une valeur haute aux évènements corrélés et une valeur négative si un objet n'a aucun lien avec le contexte considéré. On utilise généralement plutôt la variante positive ou PPMI (*Positive PMI*, voir équation 5.3) dans laquelle on remplace les valeurs de PMI négatives par zéro. La PMI comporte un biais en regard des évènements rares, nous utilisons donc également la mesure d'IDF pour introduire une forme de correction.

Enfin, afin de limiter le nombre de composants dans les vecteurs finaux, ce qui permet d'obtenir une ressource utilisable et conduit généralement à de meilleures performances, on utilise souvent une méthode de réduction de dimensions. Se limiter aux contextes les plus fréquents est généralement une mauvaise stratégie car les plus fréquents sont également souvent les moins informatifs. Il a été proposé d'utiliser plutôt des algorithmes de réduction de dimensions comme l'algorithme d'analyse en composantes principales (*Principal Component Analysis*, PCA) (PEARSON, 1901) qui permet d'effectuer une projection linéaire des données dans un espace généralement de plus basse dimension dans lequel les variables sont décorrélées tout en conservant l'essentiel de la variance originelle des données. Plus le nombre de dimensions conservées est grand, plus on conserve de variance. En appliquant une PCA, on obtient des variables décorrélées donc moins redondantes. On conserve en général seulement les premières dimensions ou axes de la PCA en regard de la variance qu'ils permettent de conserver. Lors de la construction d'un plongement lexical distributionnel, les auteurs proposent des versions comportant des nombres différents de dimensions.

5.6.2 Construction du plongement lexical

Nous construisons la représentation distributionnelle à partir des données artificielles *Bllip* présentées dans le chapitre précédent, le corpus PDTB étant de taille trop restreinte pour permettre la construction d'une telle ressource : il faut en effet disposer de suffisamment de données pour rencontrer les termes suffisamment souvent et donc obtenir une bonne estimation des informations contextuelles. Avec le corpus artificiel construit à partir du *Bllip*, nous disposons d'un texte

semi-annoté : nous avons identifié les connecteurs et leurs arguments. Contrairement à BIRAN et MCKEOWN (2013), nous utilisons donc une segmentation automatique définie sur des modèles et des heuristiques plutôt que de nous restreindre à un motif intra-phrastique sans considération du connecteur courant. Nous espérons ainsi limiter le bruit. Nous utilisons la tokenisation en mots telle qu'annotée dans le corpus *Bllip*. Nous laissons la casse inchangée, la présence de certains mots en début de phrase, donc conservant une majuscule initiale, pouvant être pertinent. Nous conservons également la ponctuation et n'effectuons aucune transformation particulière des données.

On note $\mathcal{V}$ l'ensemble des n mots présents dans les arguments des connecteurs. On note $\mathcal{C}$ l'ensemble des p connecteurs [9], c'est-à-dire l'ensemble des contextes. On construit la matrice $\mathbf{F}$ de taille $n \times p$ en calculant pour chaque élément de $\mathcal{V}$ sa fréquence d'apparition avec chaque élément de $\mathcal{C}$. On note $f_{i,j}$ la fréquence du mot $w_i \in \mathcal{V}$ apparaissant dans l'un des arguments du connecteur $c_j \in \mathcal{C}$, donc le nombre de fois où le mot w_i apparaît avec c_j.

Nous testons deux formes de pondération des fréquences brutes : la mesure TF normalisée et la mesure PPMI. La mesure TF normalisée pour un mot w_i et un connecteur c_j est définie par la formule (5.1).

$$\text{TF}_{i,j} = \frac{f_{i,j}}{\sum_{k=1}^{n} f_{k,j}} \tag{5.1}$$

La mesure PMI pour un mot w_i et un connecteur c_j (formule (5.2)) est définie à partir de la probabilité estimée $p_{i,j}$ que le mot w_i apparaisse dans le contexte de c_j et des probabilités estimées $p_{i,*}$ et $p_{*,j}$ qui correspondent respectivement aux fréquences relatives de w_i et de c_j. La mesure PPMI, équation (5.3), correspond simplement à ignorer les valeurs négatives de PMI.

$$p_{i,j} = \frac{f_{i,j}}{\sum_{k=1}^{n}\sum_{l=1}^{p} f_{k,l}}$$
$$p_{i,*} = \frac{\sum_{j=1}^{p} f_{i,j}}{\sum_{k=1}^{n}\sum_{l=1}^{p} f_{k,l}}$$
$$p_{*,j} = \frac{\sum_{i=1}^{n} f_{i,j}}{\sum_{k=1}^{n}\sum_{l=1}^{p} f_{k,l}}$$
$$\text{PMI}_{i,j} = \log\left(\frac{p_{i,j}}{p_{i,*}\, p_{*,j}}\right) \tag{5.2}$$
$$\text{PPMI}_{i,j} = \begin{cases} \text{PMI}_{i,j} & \text{si PMI}_{i,j} > 0 \\ 0 & \text{sinon} \end{cases} \tag{5.3}$$

Nous normalisons ces deux mesures en utilisant la mesure IDF, comme le font BIRAN et MCKEOWN (2013), définie pour un mot w_i par la formule (5.4). Elle correspond au logarithme de l'inverse de la proportion de connecteurs ou contextes associés au terme w_i.

$$\text{IDF}_i = \log\left(\frac{p}{\sum_{k=1}^{p} f_{i,k}}\right) \tag{5.4}$$

La normalisation correspond à la multiplication de la mesure TF ou PPMI par la valeur IDF pour chaque terme. On obtient donc finalement les plongements lexicaux qui correspondent à des matrices réelles $\mathbf{M}_{\text{TF}}$ et $\mathbf{M}_{\text{PPMI}}$ de dimension $n \times p$ définies par les formules suivantes en utilisant les définitions en (5.1), (5.3) et (5.4) :

9. Nous disposons de 96 connecteurs, voir section 4.4.2.2.

$$m_{\text{TF},i,j} = \text{TF}_{i,j} * \text{IDF}_i \text{ avec } m_{\text{TF},i,j} \in \mathbf{M}_{\text{TF}} \tag{5.5}$$

$$m_{\text{PPMI},i,j} = \text{PPMI}_{i,j} * \text{IDF}_i \text{ avec } m_{\text{PPMI},i,j} \in \mathbf{M}_{\text{PPMI}} \tag{5.6}$$

Dans ces matrices, la i^e ligne correspond au vecteur de dimension p du i^e mot du vocabulaire $\mathcal{V}$. La j^e colonne est un vecteur correspondant à un connecteur particulier. Les mots sont donc représentés dans l'espace des connecteurs.

Nous disposons d'environ une centaine de connecteurs donc les représentations sont déjà dans un espace suffisamment bas pour constituer des ressources utilisables. Il a cependant été montré qu'effectuer une réduction de dimensions pouvait améliorer les performances, TURNEY et PANTEL (2010) rapportent ainsi que ce type de méthode permet de capturer les dimensions latentes entre les mots et leur contexte et de forcer une correspondance plus importante entre eux. Cela permet également d'effectuer une forme de réduction du bruit, les premiers axes capturant la majorité du signal et les derniers correspondant plutôt au bruit dans les données. Enfin, notons que l'application d'un algorithme de réduction de dimensions permet de diminuer l'éparpillement en menant à une matrice dense. Nous testons des versions en effectuant une analyse en composant principal (PCA) ou non. Les versions avec PCA comportent 2, 5, 10 et 50 composants conservés. Le premier composant ne conserve que $11,3\%$ de la variance des données, avec 5 dimensions, on conserve $36,6\%$ de la variance, avec 10 dimensions, on conserve $56,2\%$ de la variance et avec 50 dimensions, on conserve $95,3\%$ de la variance.

5.6.3 Expériences en binaire au niveau 1

Nous présentons les résultats obtenus en binaire dans le tableau 5.11 en reprenant les résultats obtenus précédemment pour les systèmes de référence (*One-hot* $\otimes$ et *One-hot* $\oplus$). Nous rapportons également les scores obtenus par BIRAN et MCKEOWN (2013) car ces auteurs utilisent une stratégie similaire : ils construisent également une forme de plongement lexical à partir de données artificielles mais l'utilisent en définissant un nouveau trait par connecteur pour chaque instance implicite qui correspond à la similarité cosinus entre les paires associées au connecteur et celles contenues dans l'instance implicite.

Représentation	*Temporal*		*Contingency*		*Comparison*		*Expansion*	
	P	F1	P	F1	P	F1	P	F1
(BIRAN et MCKEOWN, 2013) TF-IDF	-	19,5	-	44,0	-	23,0	-	**66,5**
(BIRAN et MCKEOWN, 2013) PMI-IDF	-	16,0	-	39,0	-	24,4	-	62,2
(BIRAN et MCKEOWN, 2013) TF-IDF Stop-List	-	17,0	-	44,3	-	23,8	-	65,3
One-hot $\otimes$	23,6	21,1	41,4	50,4	26,3	34,8	62,5	59,4
One-hot $\oplus$	17,9	23,0	41,4	51,3	25,0	34,1	62,3	59,0
Best *Bllip* TF-IDF $\oplus$	14,6	24,7	41,6	**52,3**	24,0	35,0	56,9	$63,6^{**}$
Best *Bllip* PPMI-IDF $\oplus$	14,5	24,5	40,6	51,0	21,8	31,8	64,2	$63,6^{**}$
Best *Bllip* TF-IDF $\otimes$	14,6	20,7	40,1	50,6	24,0	34,4	62,5	62,8
Best *Bllip* PPMI-IDF $\otimes$	16,9	**24,9**	38,2	47,9	24,8	**35,5**	64,7	$63,2^{*}$

Table 5.11.: Modèles utilisant tous les mots sur les arguments sur l'anglais en binaire au niveau 1 avec les représentations construites à partir du *Bllip*, précision (« P ») et F_1 par relation. Nous rapportons également les scores de (BIRAN et MCKEOWN, 2013) pour leur système n'utilisant que les paires de mots sur les arguments. * $p \leq 0.1$, ** $p \leq 0.05$ comparé à *One-hot* $\otimes$ avec le t-test et Wilcoxon.

On observe comme précédemment que les meilleurs systèmes sont obtenus en utilisant des représentations denses. Les différences entre utilisation de la mesure TF ou PPMI ne sont pas très importantes, tandis que BIRAN et MCKEOWN (2013) trouvaient des écarts plus forts entre les deux

scores. Comme pour ces auteurs, nous ne pouvons conclure sur la supériorité de l'une des mesures par rapport à l'autre pour toutes les relations sur l'ensemble d'évaluation. On observe cependant sur l'ensemble de développement que la mesure PPMI est généralement meilleure que ce soit pour la forme concaténative (système $\oplus$, voir figure 5.5) ou le schéma utilisant le produit de Kronecker (système $\otimes$, voir figure 5.6). De plus, on a une régularité sur le nombre de dimensions optimal. C'est en utilisant les versions utilisant la PCA avec 50 dimensions, donc la seule version permettant de conserver plus de 90% de la variance des données, ou les versions n'utilisant pas la PCA que l'on obtient les meilleurs scores. La seule exception correspond à la relation *Expansion* avec la forme concaténative et la mesure TF. Dans ce cas, c'est avec 5 dimensions que l'on obtient le meilleur score de F_1 avec cependant une baisse importante en précision. Le plongement lexical construit à partir du *Bllip* correspond donc à un comportement plus constant pour toutes les relations que les ressources utilisées précédemment ce qui est probablement dû au fait qu'elle est plus liée à la tâche. Nous avons également évalué la représentation fondée sur les seules têtes des arguments, représentation pour laquelle nous n'obtenons pas de scores supérieurs à ceux obtenus avec les représentations de mots existantes et qui ne permet pas de dépasser les scores obtenus en utilisant la simple concaténation (*One-hot* $\oplus$) sauf, mais de manière non significative, pour la relation *Temporal* (F_1 de $24,2\%$).

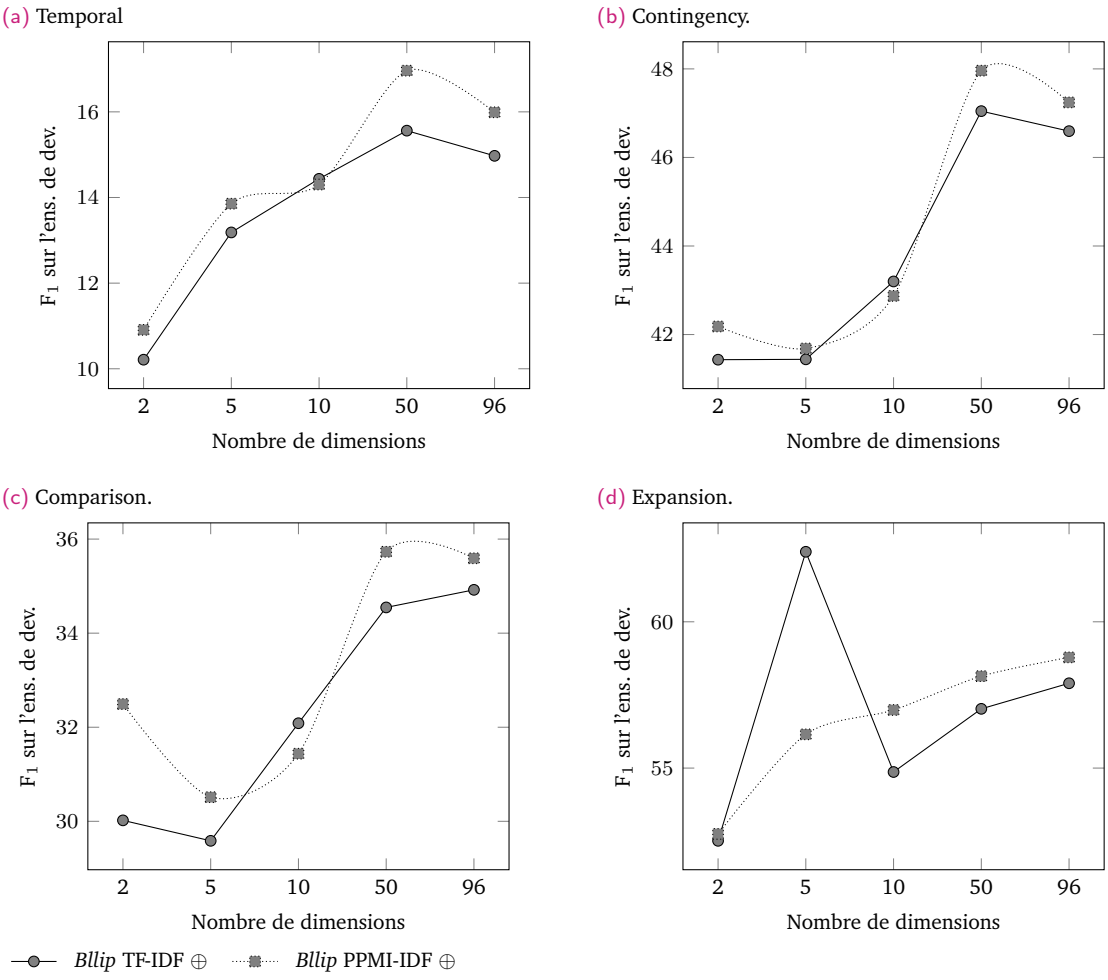


Figure 5.5.: Scores de F_1 sur l'ensemble de développement par rapport au nombre de dimensions des plongements lexicaux construits à partir du *Bllip* pour les systèmes sur l'anglais en binaire au niveau 1 utilisant la concaténation (systèmes $\oplus$). La valeur 96 correspond aux systèmes construits sans PCA.

Les résultats obtenus en termes de F_1 sont supérieurs à ceux rapportés par Biran et McKeown (2013) ce qui semble montrer que les méthodes de combinaison sont plus pertinentes que l'utilisa-

tion de traits construits sur la similarité cosinus. Ceci étant dit, nous n'obtenons pas d'amélioration significative en termes de F_1 par rapport aux systèmes de référence et les résultats sont inférieurs à ceux obtenus précédemment à part pour *Expansion*. Ceci peut être dû au fait que nous utilisons moins de données pour construire le plongement lexical que, par exemple, dans (Lebret et Collobert, 2014). Il est également possible que certains pré-traitements, comme la transformation des noms propres ou des chiffres vers un code spécifique, permettent d'améliorer les performances. De plus, notons que Biran et McKeown (2013) et Conrath et al. (2014) modélisent directement les paires de mots tandis que nous nous sommes limitée aux mots simples. Les résultats encourageants obtenus ici nous font envisager des améliorations supplémentaires avec une modélisation de ce type qui nécessitera cependant de disposer d'un ensemble plus large de données.

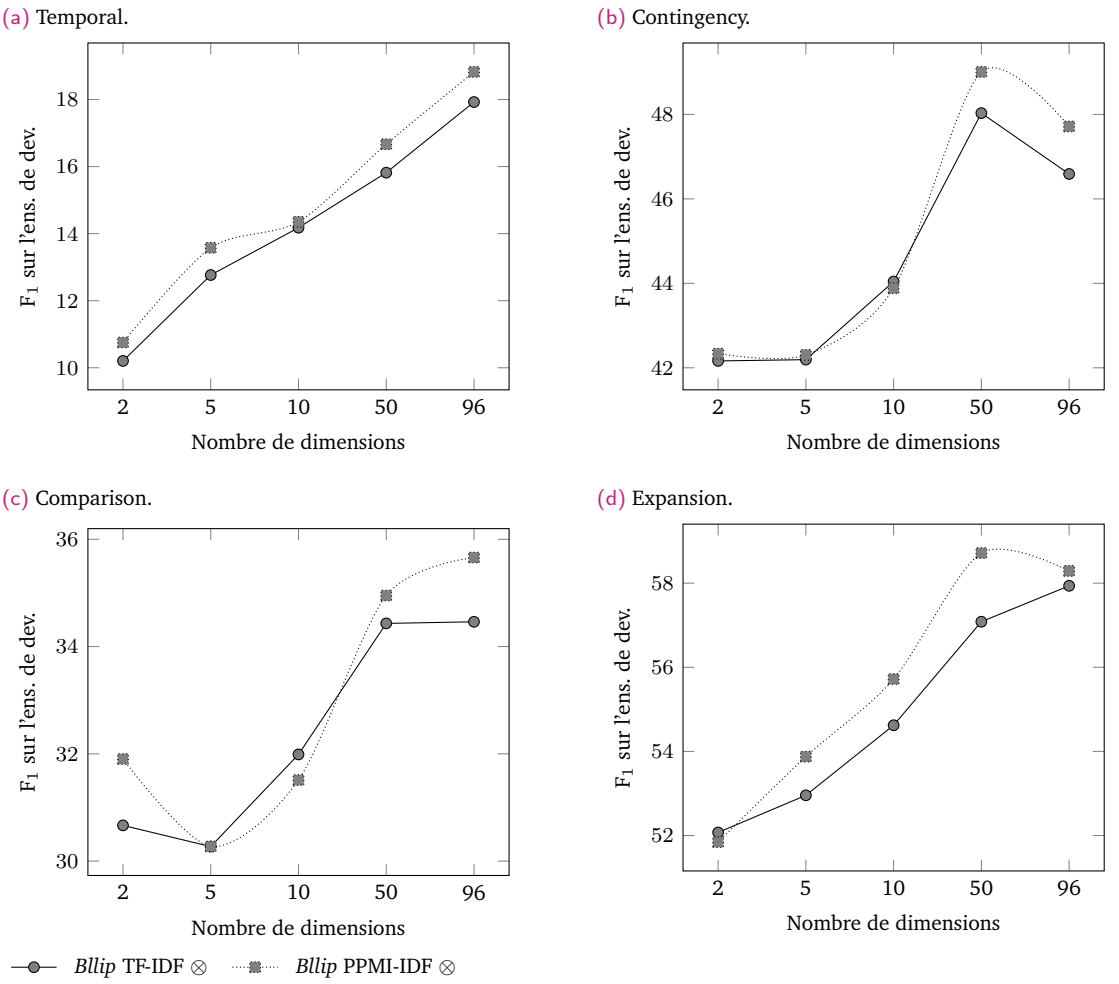


Figure 5.6.: Scores de F_1 sur l'ensemble de développement par rapport au nombre de dimensions des plongements lexicaux construits à partir du *Bllip* pour les Systèmes sur l'anglais en binaire au niveau 1 utilisant le produit de Kronecker (systèmes ⊗). La valeur 96 correspond aux systèmes construits sans PCA.

Dans le tableau 5.12, nous rapportons les scores obtenus en ajoutant des traits supplémentaires aux systèmes fondés sur les représentations construites à partir du *Bllip*. Nous reprenons également les résultats présentés précédemment correspondant à l'état de l'art (Ji et Eisenstein, 2014a ; Rutherford et Xue, 2014 ; Rutherford et Xue, 2015) ainsi que le meilleur système de Biran et McKeown (2013) qui utilise également une représentation construite sur des données explicites et des traits additionnels. Enfin, nous reportons les résultats du meilleur système obtenu avec la représentation *Bllip* seulement (Best *Bllip* seul.), avec les représentations de mots dans la section 5.5.1 (Best all tokens + traits sup.) et le système de référence correspondant aux paires de mots en forme multiplicative combinées aux traits supplémentaires (*One-hot* ⊗ all tokens + traits

sup.). Enfin, nous reprenons également les meilleurs scores obtenus au chapitre précédent avec les méthodes d'adaptation de domaine (Best Adapt.).

Représentation	*Temporal*		*Contingency*		*Comparison*		*Expansion*	
	prec	F1	prec	F1	prec	F1	prec	F1
(Biran et McKeown, 2013) Best	-	20,23	-	46,9	-	25,4	-	75,9
(Ji et Eisenstein, 2014a)	-	26,9	-	51,4	-	35,8	-	**79,9**
(Rutherford et Xue, 2015)	-	**33,3**	-	53,8	-	**41,0**	-	69,4
(Rutherford et Xue, 2014)	18,5	28,7	44,5	54,4	27,3	39,7	59,6	70,2
repr. (Rutherford et Xue, 2014) NB	24,0	28,0	49,5	52,9	28,9	37,4	53,1	70,2
repr. (Rutherford et Xue, 2014) ME	28,3	24,8	44,4	53,4	30,0	36,5	53,2	50,0
Best Adapt.	24,4	26,0	41,3	51,6	30,3	38,8	54,0	67,2
Best all tokens + traits sup.	25,8	29,3	45,2	**55,8**	26,0	36,4	65,9	61,8
One-hot $\otimes$ all tokens + traits sup.	24,6	23,3	46,3	54,4	30,6	34,3	64,9	62,6
Best *Bllip* seul.	16,9	24,9	41,6	52,3	24,8	35,5	64,2	63,6
Best *Bllip* + traits sup	22,9	30,8	42,2	53,8	26,2	37,6	65,5	60,5

Table 5.12.: Modèles utilisant tous les mots sur les arguments et des traits supplémentaires sur l'anglais en binaire au niveau 1 en se fondant sur les représentations construites à partir du *Bllip*, précision (« P ») et F_1 par relation. Nous rapportons également les meilleurs scores rapportés par (Biran et McKeown, 2013) ainsi que les scores rapportés par Rutherford et Xue (2014), les scores de Rutherford et Xue (2014) reproduits (voir section 5.5.1.5), les meilleurs résultats obtenus avec les représentations de mots utilisées précédemment dans ce chapitre (Best all tokens + traits sup.), et les meilleurs scores obtenus au chapitre précédent avec les méthodes d'adaptation (Best Adapt.). * $p \leq 0.1$, ** $p \leq 0.05$ comparé à *One-hot* $\otimes$ avec le t-test et Wilcoxon.

L'ajout des traits supplémentaires permet comme auparavant d'améliorer en général les résultats, sauf pour la relation *Expansion*. On obtient notamment une amélioration de $5,9\%$ de F_1 pour *Temporal*, de $2,1\%$ pour *Comparison* et une légère amélioration de $0,6\%$ pour *Contingency*. Comme on peut le voir dans le tableau, on obtient des résultats meilleurs que ceux obtenus précédemment (Best all tokens + traits sup.) pour *Temporal* et *Comparison*, relations pour lesquelles nous dépassons également les scores obtenus en reproduisant le système de (Rutherford et Xue, 2014). Nous ne dépassons cependant pas les scores rapportés par ces auteurs pour *Comparison*.

Ceci étant dit, encore une fois, l'ajout des traits supplémentaires ne permet pas d'amélioration significative par rapport aux systèmes utilisant uniquement les représentations de mots. Cette remarque ne vaut cependant pas pour *Temporal* qui semble particulièrement profiter des nouvelles informations alors que l'amélioration observée était beaucoup plus faible lorsque nous utilisions les représentations de mots précédentes. Par ailleurs, les scores obtenus avec ces traits supplémentaires, en utilisant les représentations précédentes ou le *Bllip*, sont similaires. Il semble donc que la représentation que nous avons construite n'encode pas certaines informations cruciales pour cette relation, une piste qu'il faudra explorer pour l'améliorer.

L'utilisation de représentations de mots, que ce soit celles utilisées précédemment, construites sans visée discursive, ou celles construites à partir du *Bllip* permet d'améliorer assez largement les scores obtenus avec les méthodes d'adaptation (Best Adapt.) pour *Temporal* et *Contingency*. Par contre, l'utilisation des données artificielles combinées aux données naturelles donne de meilleurs scores en termes de F_1 pour *Comparison*, relation pour laquelle ce système surpasse également notre reproduction de (Rutherford et Xue, 2014), et pour *Expansion*, bien que pour cette relation la précision baisse largement. Ces observations nous amènent à penser qu'une combinaison de ces stratégies pourrait conduire à des améliorations supplémentaires.

5.6.4 Expériences en multiclasse au niveau 1

Nous testons également les plongements lexicaux construits sur le *Bllip* en multiclasse au premier niveau de relation du PDTB avec la même configuration que dans les expériences précédentes. Les résultats pour les meilleurs systèmes sont présentés dans le tableau 5.13. On observe qu'ici on obtient un système qui correspond à un score de macro-F_1 significativement supérieur au système de référence utilisant les mots bruts en forme multiplicative (*One-hot* $\otimes$), la représentation traditionnellement utilisée dans les précédentes études. Cette fois-ci, les résultats sont supérieurs à ce que l'on obtenait avec les représentations de mots construites sans lien avec la tâche (au mieux $41,6\%$ de macro-F_1). Ils sont cependant inférieurs à ce que nous avions obtenu avec les méthodes d'adaptation de domaine dans le chapitre précédent (au mieux $43,8\%$). Tous les meilleurs systèmes sont obtenus avec les ressources construites sans utiliser de réduction de dimension.

Représentation	Macro-prec	Macro-F_1	Exactitude
One-hot $\otimes$	$38,6$	$39,0$	$48,6$
One-hot $\oplus$	$39,8$	$40,2$	$50,2$
Best *Embed.* $\oplus$	$41,4$	$41,6$	$50,1$
Best Adapt.	$43,0$	$\mathbf{43,8}$	$51,6$
Best *Bllip* TF-IDF $\oplus$	$40,1$	$40,1$	$50,0$
Best *Bllip* PPMI-IDF $\oplus$	$42,8$	$\mathbf{42,2}^*$	$52,5$
Best *Bllip* TF-IDF $\otimes$	$36,4$	$41,4$	$51,0$
Best *Bllip* PPMI-IDF $\otimes$	$38,5$	$38,9$	$48,2$
Best *Bllip* + traits sup.	$42,3$	$\mathbf{42,8}^*$	$51,7$

Table 5.13.: Modèles utilisant tous les mots sur les arguments sur l'anglais en multiclasse au niveau 1, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »), et scores d'exactitude de chaque système. Nous reprenons également les meilleurs scores obtenus avec les représentations de mots dans la section précédente (Best *Embed.* $\oplus$) et les meilleurs scores obtenus avec les méthodes d'adaptation (Best Adapt.). * $p \leq 0.1$ comparé à *One-hot* $\otimes$ avec le t-test et Wilcoxon.

Nous rapportons les scores par relation pour notre meilleur système construit à partir du *Bllip* dans le tableau 5.14c. Nous reprenons également, pour comparaison, les scores obtenus avec le système de référence *One-hot* $\otimes$ dans le tableau 5.14a, le meilleur système obtenu avec les ressources de mots utilisées précédemment dans le tableau 5.14b, le meilleur système obtenu avec les méthodes d'adaptation du chapitre précédent dans le tableau 5.14e, ainsi que le meilleur système rapporté par RUTHERFORD et XUE (2015) dans le tableau 5.14f. L'utilisation du plongement construit à partir du *Bllip* nous permet d'obtenir les meilleurs scores en termes de F_1 pour *Expansion* parmi l'ensemble de nos systèmes et d'améliorer l'état de l'art pour les relations *Temporal* et *Contingency*. Par contre, les représentations de mots ne permettent pas d'améliorer l'identification de la relation *Comparison*, contrairement aux stratégies fondées sur des méthodes d'adaptation de domaine (Best Adapt.).

Nous avons également évalué la représentation utilisant uniquement les têtes des arguments. Nous observons comme précédemment que l'utilisation des représentations de mots permet d'améliorer les performances du système de référence *One-hot* $\otimes$ correspondant à des performances très basses en se limitant aux têtes avec au mieux un score de macro-F_1 de $36,5\%$ (forme concaténative avec la mesure PPMI-IDF). Les performances restent cependant largement inférieures à celle d'un système utilisant tous les mots des arguments même avec la représentation la plus éparpillée traditionnelle.

(a) *One-hot* $\otimes$.

Rel	P	R	F_1
Temp	15,6	25,0	19,2
Cont	44,0	41,3	42,6
Comp	32,1	34,9	33,4
Expa	62,8	58,6	60,6

(b) Best *Embed.* $\oplus$.

Rel	P	R	F_1
Temp	19,0	39,7	25,7
Cont	47,5	44,9	46,2
Comp	35,1	31,5	33,2
Expa	63,9	58,8	61,2

(c) Best *Bllip* PPMI-IDF $\oplus$.

Rel	P	R	F_1
Temp	23,0	42,6	**29,9**
Cont	49,6	44,9	**47,1**
Comp	35,9	22,6	27,7
Expa	62,8	65,3	64,0

(d) Best *Bllip* + traits sup.

Rel	P	R	F_1
Temp	23,7	33,8	27,9
Cont	46,7	46,0	46,3
Comp	35,0	33,6	34,3
Expa	63,7	61,5	62,6

(e) Best Adapt.

Rel	P	R	F_1
Temp	22,0	38,2	28,0
Cont	44,9	46,4	45,6
Comp	44,4	35,6	**39,5**
Expa	63,9	60,4	62,1

(f) (RUTHERFORD et XUE, 2015).

Rel	P	R	F_1
Temp	38,5	9,1	14,7
Cont	49,3	39,6	43,9
Comp	44,9	27,6	34,2
Expa	61,4	78,8	**69,1**

Table 5.14.: Scores de précision (« P »), rappel (« R ») et F_1 par relation pour les systèmes sur l'anglais en multiclasse au niveau 1 : système de référence *One-hot* $\otimes$ et meilleur système fondé sur le plongement construit à partir du *Bllip* (Best *Bllip* PPMI-IDF $\oplus$) utilisant tous les mots sur les arguments. Nous reprenons également les scores du meilleur système obtenus avec les représentations de mots existantes présentés dans la section précédente (Best *Embed.* $\oplus$), les scores du meilleur système obtenu avec les méthodes d'adaptation de domaine présentés dans le chapitre précédent (Best Adapt.) et les scores du meilleur système présenté dans (RUTHERFORD et XUE, 2015).

5.6.5 Expériences en multiclasse au niveau 2

Si nous avons pu obtenir au niveau 1 de relation des améliorations grâce à la représentation construite à partir du *Bllip*, au niveau 2, cette représentation conduit à des baisses de performance importantes comme le montrent les scores obtenus et présentés dans le tableau 5.15 (le système correspondant aux meilleurs scores dans la section précédente, avec les représentations de mots construites sans visée discursive, était *One-hot* $\oplus$). Ceci montre que cette ressource n'est pas adaptée à des distinctions fines. Pour améliorer la représentation à ce niveau, nous pensons qu'il faudrait définir les contextes des mots selon leur position par rapport au connecteur, droite ou gauche, ce qui permettrait de distinguer des relations asymétriques comme *Result* et *Explanation*. Les performances en termes de macro-F_1 sont généralement très inférieurs à ce que nous avions obtenu au mieux avec les méthodes d'adaptation de domaine (Best Adapt.), performances cependant similaires à celles obtenues avec le modèle de référence.

En utilisant uniquement les têtes des arguments, on obtient toujours une amélioration par rapport au système de référence *One-hot* $\otimes$ avec cependant des scores qui demeurent inférieurs à ceux obtenus avec *One-hot* $\oplus$ ou avec les représentations utilisées dans la section précédente avec au mieux $13,6\%$ de macro-F_1 (système par concaténation utilisation la PPMI-IDF).

5.7 Conclusion du chapitre

Nous avons présenté dans ce chapitre des stratégies visant à transformer une représentation surfacique des instances de relations discursives implicites en une représentation plus dense, donc

Représentation	Macro-prec	Macro-F_1	Exactitude
One-hot $\otimes$	32, 5	23, 7	38, 4
One-hot $\oplus$	23, 4	24, 1	35, 5
Best Adapt.	27, 3	24, 1	36, 7
Best *Bllip* TF-IDF $\oplus$	20, 4	17, 6	28, 7
Best *Bllip* PPMI-IDF $\oplus$	24, 4	17, 2	30, 4
Best *Bllip* TF-IDF $\otimes$	19, 3	18, 6	27, 7
Best *Bllip* PPMI-IDF $\otimes$	17, 0	17, 3	27, 5
Best *Bllip* + traits sup.	21, 1	21, 1	33, 6

Table 5.15.: Modèles utilisant tous les mots sur les arguments sur l'anglais en multiclasse au niveau 2, scores de F_1 et de précision macro-moyennés (« macro-F_1 » et « macro-prec »)), et scores d'exactitude de chaque système. Nous reprenons les meilleurs scores obtenus avec les méthodes d'adaptation de domaines au chapitre précédent (Best Adapt.).

moins sujette à l'éparpillement, et dont les dimensions capturent une information pertinente pour la tâche. La transformation est effectuée en utilisant des représentations de mots acquises automatiquement à partir de données brutes, elle ne requiert donc pas un travail d'annotation manuelle et peut être utilisée virtuellement pour toute langue. L'utilisation de ces représentations nécessite par contre de définir des méthodes pour obtenir un vecteur composite représentant une instance de relation discursive, c'est-à-dire une paire de segments textuels. Nous avons proposé d'utiliser la simple somme sur les représentations de chaque mot pour représenter un argument et des opérations sur les vecteurs obtenus, concaténation et produit de Kronecker, pour les combiner tout en conservant l'information d'ordre. Nous avons également cherché à savoir si la tête sémantique des arguments pouvait être un bon indicateur de la relation discursive. Enfin, nous avons mené une comparaison entre des encodages ont-hot des données de densité variable, soit en se fondant directement sur les mots bruts soit en utilisant une représentation clusterisée, et des représentations denses à valeur réelle, en utilisant une représentation de type distributionnel ou distribué. Les représentations utilisées pour l'anglais sont librement distribuées mais n'ont pas été construites avec une visée discursive. Nous avons donc également évalué la possibilité de construire un plongement lexical adapté à la tâche d'identification des relations discursives en utilisant les connecteurs.

Nos expériences ont montré que l'utilisation de représentations denses menait à des améliorations significatives par rapport à la représentation sur les paires de mots dans son encodage traditionnel au niveau 1 de relation. Par contre, au niveau 2, la modélisation dense ne permet pas d'obtenir des améliorations ce qui indique que les dimensions ne reflètent pas les distinctions plus fines de ce niveau. Nous avons également trouvé que l'ajout de l'ensemble de traits obtenus à partir d'analyseurs syntaxiques et de ressources construites à la main traditionnellement utilisés n'apporte pas d'améliorations importantes au niveau 1 montrant que la majorité des informations encodées par ces traits sont fournies par les représentations de mots. Ceci permet d'envisager la construction de systèmes performants pour des langues peu dotées. Les systèmes ne se fondant que sur la tête sémantique des arguments correspondent, dans l'encodage one-hot traditionnel, à des scores très bas. Pour cette représentation simple, il est clair que les représentations de mots permettent une généralisation pertinente conduisant à une amélioration importante des scores. Ces performances sont cependant encore inférieures à celles obtenues en utilisant tous les mots sur les arguments car certaines informations cruciales ne sont pas prises en compte. Nous envisageons de déterminer quels sont précisément les autres mots qu'il faudrait inclure et comment les combiner. Quant à la ressource construite à partir des connecteurs, donc plus adaptée à la tâche, elle nous a permis d'obtenir des améliorations en multiclasse au niveau 1. Ces résultats encourageants nous amènent à envisager d'améliorer cette ressource en utilisant notamment d'autres schémas de pondération fondés directement sur les paires de mots.

Finalement, en binaire au niveau 1 nous obtenons au mieux $29,3\%$ en F_1 pour *Temporal*, $55,8\%$ pour *Contingency*, $37,6\%$ pour *Comparison* et $69,2\%$ pour *Expansion*. Pour *Comparison*, le meilleur score est obtenu en utilisant la représentation construite sur le corpus artificiel, pour les autres relations, c'est une représentation construite sans visée rhétorique qui permet d'obtenir les meilleures performances. Nous améliorons les performances obtenues par RUTHERFORD et XUE (2014) pour *Temporal* et *Contingency*, cette dernière correspondant également à une amélioration par rapport à (RUTHERFORD et XUE, 2015). Ces scores sont supérieurs à ceux obtenus dans le chapitre précédent en utilisant les données artificielles et des méthodes d'adaptation de domaine. Notons que l'approche reposant sur une transformation de la représentation à partir de représentations existantes a deux avantages importants par rapport aux stratégies d'adaptation de domaine : d'abord elle ne nécessite pas la définition (manuelle) d'heuristiques pour extraire de nouvelles données, ensuite elle inclut directement, une dimension syntaxique et sémantique dans la représentation. Il est également notable que les méthodes mises en œuvre permettent d'obtenir des modèles plus légers, c'est-à-dire computationnellement moins coûteux.

En multiclasse au niveau 1, le meilleur système correspond à $42,8\%$ de macro-F_1 ($51,7\%$ d'exactitude), score ici légèrement inférieur au meilleur système fondé sur les données artificielles (macro-F_1 de $43,8\%$). Nous obtenons également des améliorations pour la majorité des relations par rapport à (RUTHERFORD et XUE, 2015). Ces performances sont obtenues en utilisant la représentation construite à partir des données artificielles, l'utilisation d'une représentation construite sans visée discursive menant au mieux à un score de macro-F_1 de $41,6\%$, montrant la pertinence de cette représentation. Au niveau 2, nous obtenons au mieux $24,1\%$ de macro-F_1 (exactitude de $35,5\%$), score qui demeure inférieur au meilleur système de référence présenté dans le chapitre 3. Ces résultats nous amènent à envisager de combiner les deux stratégies présentées dans cette thèse de manière plus directe, en transformant également les données artificielles à partir des représentations de mots. Il est en effet possible que les améliorations obtenues par RUTHERFORD et XUE (2015) viennent du fait que la transformation des données vers un espace plus dense permet de diminuer l'écart en termes distributionnels entre les deux ensembles de données. Enfin, il est important de noter que nous avons souvent trouvé que la meilleure représentation dépendait de la relation. Nous envisageons donc de mettre en œuvre des stratégies de combinaison des représentations afin d'obtenir un seul système performant pour toutes les relations.

Conclusion

6

Sommaire

6.1 Résultats présentés

Nous avons d'abord étudié la possibilité d'utiliser des données explicites pour améliorer l'identification des relations implicites. Nous avons vu que cette stratégie repose sur deux hypothèses, la redondance du connecteur dans les instances explicites et la similarité entre données explicites et implicites. Cependant, ces hypothèses sont trop fortes : un connecteur n'est pas toujours redondant et les deux types de données sont différents que ce soit au niveau des entrées du système — les paires de segments textuels — des sorties — les relations — ou des paires d'entrée-sortie — donc l'association entre une relation et une paire de segments. Nous avons proposé de voir ces difficultés comme relevant d'une situation d'apprentissage avec données non identiquement distribuées ce qui nous a amenée à proposer des stratégies inspirées du cadre général de l'adaptation de domaine.

Nous avons mis en œuvre ces stratégies sur les données françaises issues du corpus ANNODIS et les données anglaises les plus utilisées sur la tâche d'identification des relations implicites, le corpus du PDTB. Pour ces deux langues, nous avons construit des corpus de données artificielles, donc des instances de relations explicites dans lesquelles le connecteur est supprimé afin de les rendre plus similaires aux données implicites. Ces corpus artificiels sont soit acquis automatiquement à partir d'heuristiques et/ou de modèles et de données brutes, soit également manuellement dans le cas de l'anglais. Sur les deux langues, nous avons pu observer que la construction d'un modèle des implicites sur les seules données artificielles conduisait à des performances basses, inférieures à celles obtenues avec un entraînement sur les seules données naturelles disponibles même quand ces dernières étaient pourtant en bien moins grande quantité. Ces résultats reflètent les différences en termes distributionnels et justifient la mise en place de stratégie permettant de guider le modèle vers la distribution des données naturelles.

Sur le français, les stratégies inspirées de l'adaptation de domaine nous ont permis d'obtenir, pour un ensemble de quatre relations, des améliorations significatives par rapport à un entraînement sur les seules données naturelles. Nous avons observé la même tendance sur l'anglais au niveau 1 de relation, donc pour les quatre grandes classes, mais pas au niveau 2 ce qui pourrait indiquer que ces distinctions fines sont plus spécifiques à chaque type de données ou qu'un processus de sélection particulier doit être mis en place à ce niveau notamment pour gérer le bruit qui a plus d'impact lorsque le nombre de classes est élevé.

En général, nous avons pu observer que la sélection d'exemples, cruciale dans la configuration sur le corpus français, n'apporte pas ou peu d'améliorations sur l'anglais en utilisant le corpus artificiel manuel. Nous avons également noté que, dans les deux langues, certaines relations profitent de l'apport direct des données artificielles (comme *Contrast* pour le français ou *Temporal* pour l'anglais) tandis que d'autres relations nécessitent une combinaison plus indirecte. Ces conclusions

nous amènent à penser que cette stratégie est plus adaptée pour certaines relations, pour lesquelles le connecteur est probablement moins essentiel, et qu'une étude supplémentaire des données doit être menée pour comprendre, pour les autres relations, si le problème vient de la modélisation utilisée ou de la méthode d'adaptation. Nous avons également vu que si la représentation plus riche utilisant à la fois des informations lexicales et sémantiques conduit en général aux meilleures performances, les stratégies d'adaptation permettent également d'améliorer les performances d'un système fondé sur une représentation surfacique correspondant aux paires de mots ce qui accrédite l'hypothèse originelle formulée par MARCU et ECHIHABI (2002). Finalement les meilleurs systèmes sont obtenus avec la stratégie consistant à utiliser les prédictions du modèle artificiel comme indices supplémentaires, l'un des meilleurs modèles de référence en adaptation de domaine, ou la pondération des données manuelles qui permet de diminuer l'influence des données artificielles donc de guider le modèle vers la bonne distribution. Ces résultats montrent l'adéquation du cadre de l'adaptation de domaine pour gérer les différences entre les deux ensembles de données.

Dans un second temps, nous nous sommes intéressée à la représentation surfacique des données correspondant aux paires de mots dans le produit cartésien sur les arguments d'une relation implicite. Nous avons postulé que le problème de cette représentation était son éparpillement, induit par la modélisation one-hot traditionnellement utilisée, qui entraîne des problèmes de généralisation. Bien que la présence de certaines paires de mots semble effectivement constituer un indice important voire suffisant d'une relation, leur rareté ne permet pas de construire des modèles robustes.

Nous avons donc proposé d'utiliser des représentations de mots construites ou apprises à partir de données brutes avec la visée d'associer à chaque mot une représentation plus dense dont les dimensions reflètent des caractéristiques paradigmatiques qui lui sont associées. Le problème est alors de construire un vecteur représentant une instance implicite, donc une paire de segments textuels, à partir de représentations sur les mots. Dans ce cadre nous avons mis en œuvre des stratégies de combinaison reposant sur une concaténation ou sur le produit des vecteurs correspondant à chaque segment, représenté par la somme des vecteurs représentant chacun des mots, avec la visée de conserver l'information d'ordre entre les mots.

Les représentations de mots utilisées dans un premier temps ont été construites sans visée discursive. Nous avons testé des représentations obtenues par clustering et des représentations distributionnelles et distribuées : les premières correspondent encore à une modélisation one-hot tandis que les secondes permettant d'obtenir des vecteurs denses à valeurs réelles. Ces représentations encodent des caractéristiques syntaxiques et sémantiques des mots. Les expériences menées sur l'anglais ont montré que la modélisation dense des données induite par ces représentations permettait d'obtenir des améliorations significatives au niveau 1 de relation du PDTB. En particulier, les systèmes mis en place permettent d'atteindre ou de dépasser l'état de l'art pour la majorité des relations. De plus, l'utilisation de représentation de mots semble rendre inutile l'utilisation de traits supplémentaires ce qui induit que le jeu de traits traditionnel doit être repensé afin d'identifier les informations manquantes. Par contre, les expériences menées au niveau 2 ont permis de conclure que les représentations et les modes de combinaison devaient être modifiés pour pouvoir rendre compte des distinctions plus fines de ce jeu de relations. Nous avons ensuite construit une représentation distributionnelle où un mot est associé à une dimension rhétorique à travers les connecteurs. Cette idée rejoint les stratégies mises en place dans le chapitre précédent au sens où nous utilisons des données explicites pour apprendre une représentation censée être pertinente pour les implicites. Nous avons également obtenu des améliorations importantes au niveau 1 notamment pour certaines relations tandis que les systèmes construits au niveau 2 présentent des résultats très dégradés par rapport à la simple utilisation de paires de mots bruts et ce malgré l'adaptation de la ressource à la tâche

6.2 Perspectives

Dans le cadre de la tâche d'identification des relations discursives implicites, nos perspectives les plus immédiates concernent l'extension du système développé sur le français à d'autres relations, l'évaluation des méthodes reposant sur des représentations de mots pour cette langue et l'utilisation des données explicites en cours d'annotation dans le cadre du corpus *French Discourse Treebank* pour constituer un corpus artificiel non bruité [1]. Bien sûr, lorsque ce second corpus sera entièrement annoté, nous développerons un système d'identification des relations implicites à partir de ce corpus plus large. Nous voulons également évaluer l'impact de la modélisation des données à partir des représentations de mots sur les méthodes d'adaptation de domaine. Il sera intéressant d'évaluer si son utilisation permet de réduire les différences entre les deux types de données ou de mieux identifier les exemples artificiels les plus déviants à travers de mesures de divergence. Enfin, comme nous obtenons souvent les meilleures performances pour chaque relation avec des systèmes différents, nous voulons tester des méthodes d'ensemble consistant à combiner plusieurs modèles et par exemple à les faire voter (DIETTERICH, 2000).

Une autre direction que nous voulons suivre concerne la mise en œuvre de stratégies plus sophistiquées pour combiner données naturelles et artificielles. Les résultats obtenus dans cette thèse nous encouragent à poursuivre nos investigations dans le cadre de l'adaptation de domaine pour lequel de nombreuses méthodes ont été développées. Les plus simples à mettre en œuvre sont les méthodes consistant en une pondération des instances fondée sur un modèle d'identification de l'appartenance à un domaine (ZADROZNY, 2004 ; BICKEL et SCHEFFER, 2007 ; BICKEL et al., 2007) ou sur des méthodes de boosting (DAI et al., 2007). La pondération d'instances a l'attrait, par rapport à la sélection que nous avons effectuée et qui a montré son importance, de conserver toutes les instances et d'affecter également les instances naturelles. En effet, si nous considérons que la sélection donc la suppression d'instances parmi les données naturelles comme effectuée par WANG et al. (2012) n'est pas une bonne solution, au sens où le nombre de données manuellement annotées est déjà bien assez limité, il est possible que certaines instances implicites soient plus ou moins importantes. D'autres fonctions de pondération pourraient être utilisées en s'inspirant des mesures proposées dans le cadre de la détection de données déviantes (DAELEMANS et al., 1999) ou de celles mises en œuvre dans un cadre de sélection par RUTHERFORD et XUE (2015). Nous ne sommes pas totalement convaincue par la méthode proposée par ces derniers au sens où elle nécessite de disposer d'un corpus de données implicites pour lesquelles un connecteur est annoté ce qui correspond à une stratégie dans le *Penn Discourse Treebank* destinée à faciliter l'annotation mais cette information risque de ne pas être disponible pour toutes les langues. Il est cependant clair que les critères mis en œuvre par ces auteurs sont très intéressants puisqu'ils se fondent sur la divergence entre les deux types de données et la redondance du connecteur, donc les problèmes qui nous ont conduit à nous intéresser au cadre de l'adaptation de domaine. Il faudra chercher à construire des critères du même type ne nécessitant pas de connecteurs implicites. Enfin, nous voulons également tester des méthodes d'adaptation de domaine reposant sur une modification de l'espace de représentation ou une combinaison plus fine des modèles construits sur chaque ensemble de données (BLITZER et al., 2006 ; FINKEL et MANNING, 2009).

Une autre ligne de recherche concerne le type de données ajoutées. Dans cette thèse, nous nous sommes intéressée à l'ajout de données explicites. Une autre possibilité intéressante concerne la possibilité d'ajouter des données annotées dans le cadre d'un autre corpus discursif pour les langues qui disposent de plusieurs jeux de données, ce qui est le cas au moins pour l'anglais, l'espagnol, l'allemand et prochainement pour le français. Bien sûr, dans ce cadre, l'idée serait d'ajouter d'autres

1. Rappelons que les connecteurs sont déjà annotés, l'identification des arguments et des relations est en cours (STEINLIN et al., 2015).

données implicites avant d'envisager la possibilité d'utiliser des données artificielles manuellement annotées supplémentaires, puisque cela nous permettrait de rester, à première vue, dans un cadre classique d'apprentissage supervisé. Cependant, notons d'abord que les corpus ne contiennent pas tous une annotation des exemples implicites, ce qui est par exemple le cas pour le corpus français ANNODIS ou pour le corpus anglais du *GraphBank*[2]. Il est par contre possible de les identifier automatiquement, notamment si l'on dispose par ailleurs de données explicites pour construire un modèle reconnaissant les connecteurs. La seconde difficulté concerne les différences en termes de genre et surtout de schémas d'annotation : nous avons évoqué les études sur des correspondances entre les jeux de relations et les similarités et différences en termes de segmentation. Le problème de la structure n'est ici pas crucial si nous restons au niveau de la tâche d'étiquetage en relations. Nous pensons que ces potentielles différences, en termes de bruit introduit par une identification automatique des relations implicites ou par des différences en termes de schémas d'annotation, correspondent également à des différences distributionnelles que nous pourrions prendre en compte à travers des méthodes d'adaptation.

Une seconde possibilité correspond à l'utilisation de données annotées manuellement et traduites automatiquement. Ainsi, nous voudrions utiliser les données anglaises pour améliorer des systèmes construits pour des langues moins bien dotées comme le français. Notons que VERSLEY (2010) a exploré cette approche dans le cadre de l'identification des connecteurs en allemand à partir d'un corpus parallèle allemand-anglais et du *Penn Discourse Treebank*. Il a ainsi pu construire un modèle relativement performant sans utiliser de données annotées en allemand. Pour le français, nous pourrions reprendre cette approche en ajoutant l'ensemble de données traduites aux données manuellement annotées qui recevraient un poids plus important afin de guider le modèle dans la bonne direction. Bien sûr, cette approche nous confronte une fois encore à des données bruitées et à des différences en termes de schémas d'annotation. Enfin, une autre piste intéressante serait l'utilisation de corpus de paraphrases ou de systèmes de génération automatique de paraphrases, ceci avec la visée une fois encore d'augmenter la taille du corpus de données manuellement annotées et de diversifier les exemples.

Concernant l'utilisation des représentations de mots, les améliorations à apporter visent d'abord les méthodes de combinaison utilisées. Nous avons mis en œuvre dans cette thèse des stratégies relativement simples notamment en ce qui concerne la transformation d'un segment en vecteur c'est-à-dire en utilisant la somme sur les mots, donc en considérant chaque argument comme un sac de mots. Nous envisageons d'évaluer des méthodes de combinaison plus fines, reposant notamment sur la structure syntaxique des arguments et l'importance des mots en utilisant les opérations proposées dans (MITCHELL et LAPATA, 2010 ; BLACOE et LAPATA, 2012). Une autre façon d'envisager cette question consiste à comprendre quelles informations manquent à la modélisation fondée sur les seules têtes sémantiques. Il manque clairement des informations cruciales à cette représentation notamment les informations temporelles apportées par les auxiliaires et les indices de modalité et d'aspect. Il est également évident que les adverbes contribuent à l'expression des relations rhétoriques et qu'il est nécessaire d'inclure la présence éventuelle d'une négation. Nous pensons qu'il est possible d'inclure toutes ces informations directement dans la représentation vectorielle dense, sans passer par une concaténation de traits supplémentaires. Il nous faut donc construire une fonction de combinaison qui permette d'avoir des vecteurs opposés pour un évènement et sa négation et qui rende compte de la modalité ou de l'aspect. Enfin, il faudrait également pouvoir modéliser certaines connaissances du monde à travers une représentation de mots comme le lien causal entre « *tomber* » et « *pousser* ». C'était bien sûr le but de notre représentation construite à partir des connecteurs, représentation que nous voulons améliorer en testant des formes de

2. Notons que les corpus du *RST Discourse Treebank* et du *Penn Discourse Treebank* ont été annotés sur les mêmes données, il est donc en fait peu envisageable d'utiliser ces deux corpus conjointement dans le cadre de l'identification des relations, une étude sur les différences serait par contre intéressante.

prétraitements, en nous restreignant probablement à un vocabulaire plus limité et en utilisant d'autres formes de pondération des fréquences. Dans ce cadre, nous envisageons également des expériences sur l'anglais et le français en utilisant la ressource construite par CONRATH et al. (2014). Cette représentation met directement en lien des paires de verbes liés à une relation et non des mots simples comme nous l'avons fait ce qui est plus pertinent pour notre tâche. Comme seuls les verbes sont pris en compte, il sera nécessaire de combiner la représentation obtenue à d'autres informations, éventuellement en utilisant la représentation que nous avons construite. Finalement, nous voulons également évaluer la possibilité d'adapter les représentations de mots existantes que nous avons utilisées, comme celles proposées par COLLOBERT et WESTON (2008) et LEBRET et COLLOBERT (2014), à notre tâche en utilisant la méthode proposée par LABUTOV et LIPSON (2013).

D'autres améliorations, peu ou pas abordées dans cette thèse, sont à envisager pour la tâche d'identification des relations implicites, certaines pouvant par ailleurs également servir pour l'identification des explicites. D'abord, rappelons que les relations discursives sont, dans le *Penn Discourse Treebank*, organisées en hiérarchie. Plus généralement, il est toujours proposé dans les cadres théoriques ou les corpus des groupements des relations. Nous pensons que la prise en compte de cette hiérarchie, à travers des algorithmes adaptés, pourrait conduire à des améliorations. Cela pourrait notamment améliorer l'identification de relations peu représentées et/ou correspondant à des distinctions fines. Ainsi, *Contrast* et *Concession* sont deux relations marquant une forme d'opposition, qui peuvent mettre en jeu des indices similaires, mais la seconde correspond à bien moins d'exemples que la première que ce soit en implicite (219 exemples de *Concession* et 2 062 *Contrast*) ou en explicite (1 201 exemples de *Concession* et 3 844 de *Contrast*). Ce déséquilibre est en partie responsable du fait que *Contrast* est bien mieux identifiée que *Concession* dans les modèles avec des scores entre 2 et 3 fois plus bas (voir les sections 3.4.3 et 4.4.2.2 pour des scores de référence en implicite et en explicite respectivement). L'utilisation d'un simple modèle hiérarchique en cascade permettrait de mettre en jeu un modèle binaire uniquement dédié à cette distinction et donc d'obtenir une meilleure estimation des paramètres. La difficulté posée par un modèle en cascade est la nécessité de disposer d'un modèle assez performant au premier niveau de la hiérarchie, pour le corpus du *Penn Discourse Treebank* au niveau 1 de relation. Or nous avons vu que ces modèles ne correspondent pas encore à des scores très hauts pour les relations implicites. Les premières expériences menées par VERSLEY (2011) pour les relations explicites ont été peu concluantes. Cependant, nous pensons que cet aspect, en plus de ne pas avoir été exploré pour les implicites, n'a pas été suffisamment exploité. Notamment, nous envisageons de tester des algorithmes d'apprentissage hiérarchique, comme celui décrit dans (DEKEL et al., 2004) adapté au cas de classes déséquilibrées dans (DEKEL, 2009).

Une seconde question que nous avons évoquée dans la conclusion des expériences présentées dans le chapitre 5 concerne la représentation des données. Nous avons vu que les représentations de mots nous permettaient d'obtenir des systèmes dont les performances sont très peu améliorées par l'ajout des autres traits traditionnellement utilisés dans les études existantes. Ceci nous semble révéler qu'une nouvelle étude sur la représentation des instances implicites est nécessaire, une étude comme celle menée par PARK et CARDIE (2012). Nous l'avons dit, les systèmes actuels utilisent énormément de ressources et construisent une représentation de grande dimensionnalité, il est cependant clair que ces systèmes peuvent être simplifiés, les représentations de mots rendant inutiles certains traits. Cependant, les performances sont encore basses ce qui implique probablement qu'il nous manque encore certaines informations, informations que l'on pourra acquérir en améliorant les représentations de mots existantes c'est-à-dire en les adaptant à la tâche. Mais il faudra également évaluer l'impact d'autres ressources construites à la main et peu ou pas exploitées jusqu'à présent comme *Wordnet* (FELLBAUM, 1998), *FrameNet* (JOHNSON et al., 2002), *PropBank* (KINGSBURY

et PALMER, 2002), *VerbNet* (KIPPER et al., 2000), *TimeBank* (PUSTEJOVSKY et al., 2003a) ou *OntoNotes* (HOVY et al., 2006). L'utilisation de ces ressources permettrait notamment d'identifier les indices manquants dans les systèmes actuels, étape qui peut être considérée comme préliminaire à la recherche de nouvelles stratégies pour améliorer les scores de manière moins supervisée.

Notre visée finale, que nous espérons mettre en œuvre rapidement, est l'intégration de nos modèles dans un système complet sur le français et sur l'anglais. Concernant l'anglais, nous commencerons par la construction d'un chunker discursif sur le *Penn Discourse Treebank* puisque c'est l'application la plus directe des systèmes que nous avons construits. Un chunker discursif correspond à l'identification de différents types de relation et de leurs arguments, et éventuellement des segments attributifs. Concernant l'identification des arguments, la seule modification aux systèmes actuels que nous envisageons pour le moment concerne une amélioration pour les cas d'arguments non consécutifs en nous inspirant des méthodes mises en œuvre dans les systèmes construits sur le *RST Discourse Treebank*. Dans ce cadre, nous voulons envisager le problème en séparant les différents types de relations plutôt qu'en les rassemblant en un groupe explicite et un groupe non explicite comme l'ont fait LIN et al. (2014). Nous envisageons de chercher à améliorer l'identification des relations explicites pour lesquelles, nous l'avons vu, certaines relations peu représentées correspondent à des scores relativement bas. Pour ce type de relation, nous voulons ajouter des méthodes pour gérer le déséquilibre des classes et tester des représentations plus riches que celles mises en œuvre traditionnellement. Il est en effet intéressant de pouvoir identifier une relation sans recourir au contenu des arguments, cela permet d'obtenir une structure partielle. Cependant, comme le but est d'identifier également ces arguments, il nous semble que, pour un système complet, s'intéresser aux informations qu'ils contiennent a du sens et pourrait probablement améliorer les performances pour des distinctions fines aujourd'hui ignorées. Nous sommes par ailleurs assez curieuse d'évaluer la possibilité d'améliorer l'identification des explicites en utilisant les implicites, donc en inversant notre schéma source-cible des données. Pour un chunker discursif, il nous faut également nous intéresser aux relations de type lexicalisation alternative et relation d'entité. Les premières ressemblent aux explicites puisqu'elles contiennent un signal fort de la relation. Nous voulons donc utiliser les explicites pour améliorer leur identification. Les relations d'entité sont plus spécifiques et nous pensons qu'elles profiteront notamment de l'utilisation d'outils de résolution de la coréférence. Enfin, nous devrons également nous intéresser aux quelques cas d'annotation d'une absence de relation que nous considèrerons dans le cadre d'un système binaire. Le problème de ces dernières est leur rareté, et une simple méthode de pondération risque de ne pas suffire, on est dans le cadre d'une rareté de l'ordre de 6 exemples positifs pour 1000, donc une rareté absolue (HE et GARCIA, 2009). Dans ce cadre, de nombreuses méthodes ont été proposées et nous envisageons de nous intéresser en particulier à celle correspondant à une génération automatique de données de la classe rare en se fondant sur des similarités entre les exemples minoritaires existants, méthode connue sous le nom de technique de sur-échantillonnage synthétique minoritaire (*synthetic minority oversampling technique*, SMOTE) (CHAWLA et al., 2002).

La construction d'un chunker discursif pourrait déjà permettre d'apporter des informations pertinentes pour différentes applications de Traitement Automatique des Langues. Il nous semble cependant que le vrai défi est la construction d'une structure discursive complète comme annotée dans le *RST Discourse Treebank* ou ANNODIS. Pour le *RST Discourse Treebank*, nous voulons évaluer la possibilité d'utiliser les modèles construits pour l'identification des relations sur le *Penn Discourse Treebank* qui contient bien plus de données ce qui rejoint notre objectif de combiner les données manuellement annotées existantes. Pour le français, nous envisageons également la possibilité d'utiliser des outils de traduction automatique et les travaux récents visant à unifier les différents types de structure (VENANT et al., 2013) afin d'utiliser les données du *RST Discourse Treebank* comme données supplémentaires pour un modèle construit sur ANNODIS. Dans les deux cas, des

méthodes d'adaptation seront nécessaires pour gérer les différences en termes de domaines et la présence de bruit dans les données.

Liste des connecteurs du PDTB

A

Nous donnons dans le tableau A.1 le nombre d'exemples et la fréquence relative des relations explicites (PDTB) et des exemples artificiels (*Bllip*) pour chaque connecteur.

Connecteur	PDTB	*Bllip*	Connecteur	PDTB	*Bllip*
but	3308 (17.92%)	516105 (17.42%)	ultimately	18 (0.1%)	918 (0.03%)
and	3000 (16.25%)	351825 (11.88%)	similarly	18 (0.1%)	206 (0.01%)
also	1746 (9.46%)	290444 (9.8%)	in other words	17 (0.09%)	20 (0.0%)
if	1223 (6.63%)	325657 (10.99%)	rather	17 (0.09%)	232 (0.01%)
when	989 (5.36%)	323607 (10.92%)	as if	16 (0.09%)	7632 (0.26%)
because	858 (4.65%)	193561 (6.53%)	meantime	15 (0.08%)	190 (0.01%)
while	781 (4.23%)	150429 (5.08%)	earlier	15 (0.08%)	68 (0.0%)
as	743 (4.03%)	36399 (1.23%)	in particular	15 (0.08%)	29 (0.0%)
after	577 (3.13%)	58548 (1.98%)	thereby	12 (0.07%)	1988 (0.07%)
however	485 (2.63%)	65651 (2.22%)	overall	12 (0.07%)	113 (0.0%)
then	340 (1.84%)	109678 (3.7%)	in contrast	12 (0.07%)	4 (0.0%)
although	328 (1.78%)	61956 (2.09%)	by comparison	11 (0.06%)	1 (0.0%)
before	326 (1.77%)	54172 (1.83%)	furthermore	11 (0.06%)	43 (0.0%)
though	320 (1.73%)	67947 (2.29%)	afterward	11 (0.06%)	2596 (0.09%)
so	263 (1.42%)	60974 (2.06%)	thereafter	11 (0.06%)	1340 (0.05%)
for example	196 (1.06%)	12 (0.0%)	except	10 (0.05%)	465 (0.02%)
meanwhile	193 (1.05%)	18920 (0.64%)	consequently	10 (0.05%)	941 (0.03%)
still	190 (1.03%)	840 (0.03%)	specifically	10 (0.05%)	131 (0.0%)
since	184 (1.0%)	22618 (0.76%)	in the end	9 (0.05%)	415 (0.01%)
in addition	165 (0.89%)	36 (0.0%)	further	9 (0.05%)	0 (0.0%)
until	162 (0.88%)	38524 (1.3%)	likewise	8 (0.04%)	1216 (0.04%)
thus	112 (0.61%)	11824 (0.4%)	next	7 (0.04%)	0 (0.0%)
instead	112 (0.61%)	21346 (0.72%)	additionally	7 (0.04%)	23 (0.0%)
indeed	104 (0.56%)	2707 (0.09%)	by then	7 (0.04%)	246 (0.01%)
yet	101 (0.55%)	7037 (0.24%)	alternatively	6 (0.03%)	60 (0.0%)
moreover	101 (0.55%)	6432 (0.22%)	simultaneously	6 (0.03%)	295 (0.01%)
or	98 (0.53%)	7509 (0.25%)	much as	6 (0.03%)	418 (0.01%)
for instance	98 (0.53%)	15 (0.0%)	as well	6 (0.03%)	13 (0.0%)
unless	95 (0.51%)	15673 (0.53%)	accordingly	5 (0.03%)	41 (0.0%)
later	91 (0.49%)	16199 (0.55%)	whereas	5 (0.03%)	1409 (0.05%)
once	84 (0.46%)	16831 (0.57%)	as though	5 (0.03%)	741 (0.03%)
in fact	82 (0.44%)	510 (0.02%)	in short	4 (0.02%)	36 (0.0%)
as a result	78 (0.42%)	6336 (0.21%)	either or	4 (0.02%)	4683 (0.16%)
separately	74 (0.4%)	1622 (0.05%)	hence	4 (0.02%)	16 (0.0%)
previously	49 (0.27%)	956 (0.03%)	on the contrary	4 (0.02%)	3 (0.0%)
nevertheless	44 (0.24%)	4999 (0.17%)	for	3 (0.02%)	0 (0.0%)
if then	38 (0.21%)	15377 (0.52%)	neither nor	3 (0.02%)	6445 (0.22%)
on the other hand	37 (0.2%)	5816 (0.2%)	till	3 (0.02%)	431 (0.01%)
finally	32 (0.17%)	910 (0.03%)	if and when	3 (0.02%)	149 (0.01%)
nor	31 (0.17%)	3393 (0.11%)	lest	2 (0.01%)	881 (0.03%)
so that	31 (0.17%)	8931 (0.3%)	in sum	2 (0.01%)	1 (0.0%)
in turn	30 (0.16%)	25 (0.0%)	as an alternative	2 (0.01%)	1 (0.0%)
by contrast	27 (0.15%)	9 (0.0%)	regardless	2 (0.01%)	17 (0.0%)
nonetheless	27 (0.15%)	4723 (0.16%)	conversely	2 (0.01%)	5 (0.0%)
therefore	26 (0.14%)	5496 (0.19%)	insofar as	1 (0.01%)	96 (0.0%)
as long as	24 (0.13%)	6679 (0.23%)	plus	1 (0.01%)	221 (0.01%)
otherwise	24 (0.13%)	958 (0.03%)	when and if	1 (0.01%)	62 (0.0%)
now that	22 (0.12%)	4582 (0.15%)	else	1 (0.01%)	22 (0.0%)
as soon as	20 (0.11%)	2717 (0.09%)	on the one hand on the other hand	1 (0.01%)	0 (0.0%)
besides	19 (0.1%)	164 (0.01%)	before and after	1 (0.01%)	4 (0.0%)

Table A.1.: Liste des 100 connecteurs du PDTB et le nombre d'exemples (fréquence) dans le PDTB et dans le corpus artificiel constitué à partir du *Bllip*. Les connecteurs discontinus sont identifiés par la présence de « .. » entre les deux parties du connecteur.

Connecteurs utilisés pour le français

B

Nous donnons dans le tableau B.1 le nombre d'exemples extraits pour chaque connecteur dans le corpus artificiel construit pour le français ainsi que les motifs utilisés et identifiés pour chacun de ces connecteurs. Nous avons regroupé les motifs en catégories générales, c'est-à-dire sans tenir compte de la ponctuation. Nous avons défini 3 motifs correspondant à une configuration interphrastique (i.e. les arguments sont deux phrases) qui diffèrent selon la position du connecteur dans la clause hôte (i.e. l'argument dont le connecteur dépend syntaxiquement, l'autre argument étant la clause conviée) : soit le connecteur est à l'initial de la clause hôte (« INTER INIT »), soit le connecteur est au sein de la clause hôte (« INTER MED »), soit le connecteur est à la fin de la clause hôte (« INTER FIN »). Pour les configurations intraphrastiques (i.e. les arguments sont deux propositions à l'intérieur d'une phrase), on retrouve cette même différenciation (« INTRA INIT », « INTRA MED » et « INTRA FIN »). Pour les cas intraphrastiques où le connecteur est à l'initial de la clause hôte, nous avons également différencié les cas où la clause hôte est au sein de la clause conviée (« INTRA INTERNE »). Nous avions également défini des motifs où la clause hôte se trouvait avant la clause conviée (clause préposée) mais nous n'avons aucun cas dans le corpus. Nous indiquons dans le tableau le nombre d'exemples par motif. Il est cependant clair que ces chiffres ne correspondent pas à la distribution réelle des connecteurs, puisque nous avons défini une heuristique qui se limite donc à certains cas et que cette heuristique est sujette à erreur. Les motifs sont définis comme suit (« A1 » correspond au premier argument, « A2 » au second argument — celui dont dépend le connecteur — et « C » au connecteur) :

— « INTER INIT » : les arguments sont deux phrases adjacentes et le connecteur est à l'initial de la clause hôte (i.e. A1. C A2),

— « INTER MED » : les arguments sont deux phrases adjacentes et le connecteur est au sein de la clause hôte (i.e. A1. A2 C A2.),

— « INTER FIN » : les arguments sont deux phrases adjacentes et le connecteur est à la fin de la clause hôte (i.e. A1. A2 C.),

— « INTRA INIT » : les arguments sont deux clauses à l'intérieur d'une même phrase et le connecteur est à l'initial de la clause hôte (i.e. A1 C A2),

— « INTRA MED » : les arguments sont deux clauses à l'intérieur d'une même phrase et le connecteur est au sein de la clause hôte (i.e. A1 A2 C A2.),

— « INTRA FIN » : les arguments sont deux clauses à l'intérieur d'une même phrase et le connecteur est à la fin de la clause hôte (i.e. A1 A2 C.),

— « INTRA INTERNE » : les arguments sont deux clauses à l'intérieur d'une même phrase et la clause hôte est située au sein de la clause invitée. Dans ce cas le connecteur est à l'initial de la clause hôte (i.e. A1 C A2 A1).

La construction de ces données artificielles est décrite dans la section 4.4.1.2.

Connecteur	Inter-phrastique			Intra-phrastique			
	INTER INIT	INTER MED	INTER FIN	INTRA INIT	INTRA INTERNE	INTRA MED	INTRA FIN
Continuation							
d'autre part	2405	1184	0	178	0	0	0
de plus	4716	91	0	212	0	0	0
en outre	1749	1698	0	216	0	3	0
en plus	1267	391	0	0	0	0	0
et puis	6033	0	0	373	0	0	0
outre que	0	0	0	28	10	0	0
par-dessus tout	1	4	0	1	0	0	0
par ailleurs	6073	564	0	0	0	6	0
parallèlement	1120	0	0	0	0	0	0
sans compter que	318	0	0	0	0	0	0
sans oublier que	158	0	0	27	0	0	0
surtout	436	0	0	0	0	0	0
Contrast							
au lieu	2	0	0	0	0	0	0
bien que	0	0	0	1197	0	0	0
ceci dit	468	0	0	0	0	0	0
ceci étant dit	21	0	0	0	0	0	0
cela dit	721	0	0	0	0	0	0
cependant	3532	7728	0	248	0	43	0
cependant que	0	0	0	21	0	0	0
comparativement	3	0	0	0	0	0	0
dire que	191	0	0	0	0	0	0
en comparaison	26	0	0	0	0	0	0
en revanche	4858	681	0	647	0	1	0
encore	10	0	0	0	0	0	0
encore que	236	0	0	57	0	0	0
et dire que	213	0	0	0	0	0	0
excepté que	0	0	0	1	0	0	0
hormis le fait que	9	0	0	1	0	0	0
hormis que	3	0	0	0	0	0	0
mais	93570	0	0	98430	0	0	0
malgré le fait que	0	0	0	3	0	0	0
malgré que	0	0	0	16	0	0	0
malgré tout	915	0	0	0	0	0	0
malheureusement	2412	0	0	0	0	0	0
mis à part le fait que	1	0	0	1	0	0	0
mis à part que	3	0	0	2	0	0	0
même si	5476	0	0	7104	0	0	0
nonobstant	3	0	0	0	0	0	0
nonobstant que	0	0	0	2	0	0	0
néanmoins	1787	4396	0	97	0	45	0
par contre	2349	0	0	0	0	0	0
pourtant	6801	324	0	480	0	0	0
quand bien même	0	0	0	42	0	0	0
quand même	32	0	0	0	0	0	0
quoique	432	0	0	91	0	0	0
réflexion faite	7	5	0	0	0	0	0
sauf que	567	0	0	124	0	0	0
sinon que	26	0	0	45	0	0	0
tout de même	17	6039	0	0	0	106	0
à ceci près que	0	0	0	23	0	0	0
à cela près que	8	0	0	0	0	0	0
à la place	33	0	0	0	0	0	0
à la réflexion	6	2	0	0	0	0	0

Connecteur	Inter-phrastique			Intra-phrastique			
	INTER INIT	INTER MED	INTER FIN	INTRA INIT	INTRA INTERNE	INTRA MED	INTRA FIN
Contrast							
à part que	0	0	0	10	0	0	0
à part ça	36	0	0	7	0	0	0
Explanation							
après tout	330	0	0	0	0	0	0
attendu que	0	0	0	15	0	0	0
car	11856	0	0	9903	0	0	0
cette fois que	0	0	0	2	0	0	0
considérant que	0	0	0	281	33	0	0
d'autant plus que	617	0	0	388	0	0	0
d'autant que	2620	0	0	1061	0	0	0
dans le sens où	3	0	0	59	0	0	0
du fait que	0	0	0	6	22	0	0
en effet	12247	1460	0	1203	0	1	0
faute de	0	0	0	730	0	0	0
le fait est que	28	0	0	0	0	0	0
par le fait que	0	0	0	1	0	0	0
parce que	2058	0	0	4431	0	0	0
puisque	0	0	0	7608	0	0	0
sachant que	0	0	0	1386	42	0	0
surtout que	636	0	0	307	0	0	0
vu que	0	0	0	69	0	0	0
à force de	0	0	0	399	0	0	0
à preuve	27	0	0	0	0	0	0
étant donné que	0	0	0	70	0	0	0
Result							
ainsi	7553	0	0	0	0	0	0
au point de	415	0	0	234	0	0	0
au point que	366	0	0	304	0	0	0
autant dire que	812	0	0	0	0	0	0
autrement dit	684	0	0	0	0	0	0
c'est pourquoi	275	0	0	0	0	0	0
comme quoi	385	0	0	14	0	0	0
de ce fait	556	0	0	0	0	0	0
de façon que	0	0	0	28	0	0	0
de sorte que	153	0	0	93	0	0	0
de telle manière que	3	0	0	1	0	0	0
donc	1049	0	0	29147	0	0	0
du coup	1951	0	0	83	0	0	0
dès lors	1119	137	0	0	0	0	0
décidément	670	0	0	0	0	0	0
en conséquence	633	38	0	0	0	3	0
jusqu'à	1912	0	0	0	0	0	0
par conséquent	256	0	0	0	0	0	0
par suite	3	0	0	0	0	0	0
preuve que	420	0	0	110	0	0	0
à ce point que	9	0	0	5	0	0	0
à force	47	0	0	0	0	0	0
à tel point que	595	0	0	184	0	0	0
à telle enseigne que	38	0	0	13	0	0	0

Table B.1.: Connecteurs utilisés pour construire le corpus artificiel pour le français.

Bibliographie

ADAM, Clémentine et Marianne VERGEZ-COURET (2012). « Exploiting naive vs expert discourse annotations : an experiment using lexical cohesion to predict Elaboration / Entity-Elaboration confusions ». In : *Proceedings of the Linguistic Annotation Workshop* (cf. p. 43).

AFANTENOS, Stergos et Nicholas ASHER (2010). « Testing SDRT's Right Frontier ». In : *Proceedings of COLING* (cf. p. 22).

AFANTENOS, Stergos, Pascal DENIS, Philippe MULLER et Laurence DANLOS (2010). « Learning Recursive Segments for Discourse Parsing ». In : *Proceedings of LREC* (cf. p. 56).

AFANTENOS, Stergos, Nicholas ASHER, Farah BENAMARA et al. (2012a). « An empirical resource for discovering cognitive principles of discourse organisation : the ANNODIS corpus ». In : *Proceedings of LREC* (cf. p. 1, 26, 42, 44, 45, 55).

AFANTENOS, Stergos, Nicholas ASHER, Farah BENAMARA et al. (2012b). « Developing a corpus of strategic conversation in the Settlers of Catan ». In : *Proceedings of the workshop on Games and NLP (GAMNLP)* (cf. p. 26).

AL-SAIF, Amal et Katja MARKERT (2010). « The Leeds Arabic Discourse Treebank : Annotating Discourse Connectives for Arabic ». In : *Proceedings of LREC* (cf. p. 27).

ALLWEIN, Erin L., Robert E. SCHAPIRE et Yoram SINGER (2000). « Reducing Multiclass to Binary : A Unifying Approach for Margin Classifiers. » In : *Journal of Machine Learning Research* 1, p. 113–141 (cf. p. 72).

ANDO, Rie K. et Tong ZHANG (2005). « A framework for learning predictive structures from multiple tasks and unlabeled data ». In : *The Journal of Machine Learning Research* 6, 1817—1853 (cf. p. 92).

ASHER, Nicholas (1993). *Reference to Abstract Objects in Discourse : A Philosophical Semantics for Natural Language Metaphysics*. T. 50. SLAP. Kluwer (cf. p. 15, 19).

ASHER, Nicholas et Alex LASCARIDES (1998). « Bridging ». In : *Journal of Semantics* 15.1, p. 83–113 (cf. p. 112).

– (2003). *Logics of Conversation*. Cambridge University Press (cf. p. 7, 14, 15, 19, 22, 37, 42, 113).

ASHER, Nicholas et Laure VIEU (2005). In : *Lingua* 115.4, p. 591–610 (cf. p. 22).

ASHER, Nicholas, Antoine VENANT, Phillipe MULLER et Stergos AFANTENOS (2011). « Complex discourse units and their semantics ». In : *Proceedings of Constraints in Discourse* (cf. p. 16).

ASR, Fatemeh et Vera DEMBERG (2013). « On the Information Conveyed by Discourse Markers ». In : *Proceedings of the Annual Workshop on Cognitive Modeling and Computational Linguistics (CMCL)* (cf. p. 113).

AXELROD, Amittai, Xiaodong HE et Jianfeng GAO (2011). « Domain Adaptation via Pseudo In-domain Data Selection ». In : *Proceedings of EMNLP* (cf. p. 124).

BEN-DAVID, Shai, John BLITZER, Koby CRAMMER et Fernando PEREIRA (2007). « Analysis of Representations for Domain Adaptation ». In : *Proceedings of Advances in Neural Information Processing Systems 20* (cf. p. 124, 126).

BENAMARA, Farah et Maite TABOADA (2015). « Mapping different rhetorical relation annotations : A proposal ». In : *Proceedings of Starsem* (cf. p. 26, 36).

BENGIO, Yoshua, Réjean DUCHARME, Pascal VINCENT et Christian JANVIN (2003). « A Neural Probabilistic Language Model ». In : *Journal of Machine Learning Research* 3, p. 1137–1155 (cf. p. 165).

BENZ, Anton et Peter KÜHNLEIN, éds. (2008). *Constraints in Discourse*. John Benjamins Publishing Company (cf. p. 15).

BERGER, Adam L., Stephen A. DELLA PIETRA et Vincent J. DELLA PIETRA (1996). « A maximum entropy approach to natural language processing ». In : 22, 39—71 (cf. p. 97).

BHATIA, Parminder, Yangfeng JI et Jacob EISENSTEIN (2015). « Better Document-level Sentiment Analysis from RST Discourse Parsing ». In : *Proceedings of EMNLP* (cf. p. 53, 65).

BICKEL, Steffen et Tobias SCHEFFER (2007). « Dirichlet-enhanced spam filtering based on biased samples ». In : *Proceedings of NIPS* (cf. p. 126, 197).

BICKEL, Steffen, Michael BRÜCKNER et Tobias SCHEFFER (2007). « Discriminative Learning for Differing Training and Test Distributions ». In : *Proceedings of ICML* (cf. p. 126, 197).

BIRAN, Or et Kathleen MCKEOWN (2013). « Aggregated Word Pair Features for Implicit Discourse Relation Disambiguation ». In : *Proceedings of ACL* (cf. p. 68, 79, 85–87, 92, 96, 183, 185–189).

BLACOE, William et Mirella LAPATA (2012). « A Comparison of Vector-based Representations for Semantic Composition ». In : *Proceedings of EMNLP-CoNLL* (cf. p. 168, 198).

BLAIR-GOLDENSOHN, Sasha, Kathleen R. MCKEOWN et Owen C. RAMBOW (2007). « Building and refining rhetorical-semantic relation models ». In : *Proceedings of NAACL HLT* (cf. p. 6, 76, 87, 89, 138, 139).

BLITZER, John, Ryan MCDONALD et Fernando PEREIRA (2006). « Domain Adaptation with Structural Correspondence Learning ». In : *Proceedings of EMNLP* (cf. p. 126, 127, 197).

BRAS, Myriam (2008). « Entre relations temporelles et relations de discours ». Habilitation à Diriger des Recherches. Université de Toulouse (cf. p. 39).

BRAUD, Chloé et Pascal DENIS (2014a). « Combining Natural and Artificial Examples to Improve Implicit Discourse Relation Identification ». In : *Proceedings of COLING* (cf. p. 108).

– (2014b). « Identifier les relations discursives implicites en combinant données naturelles et données artificielles ». In : *Traitement Automatique des Langues* 55.1, p. 135–165 (cf. p. 108).

BRAUD, Chloé et Pascal DENIS (2015). « Comparing Word Representations for Implicit Discourse Relation Classification ». In : *Proceedings of EMNLP* (cf. p. 162).

BROWN, Peter F., Peter V. DESOUZA, Robert L. MERCER, Vincent J. DELLA PIETRA et Jenifer C. LAI (1992). « Class-Based n-gram Models of Natural Language ». In : *Computational Linguistics* 18, p. 467–479 (cf. p. 7, 94, 165).

BURSTEIN, Jill, Karen KUKICH, Susanne WOLFF, Chi LU et Martin CHODOROW (1998). « Enriching Automated Essay Scoring Using Discourse Marking ». In : *Proceedings of the ACL Workshop on Discourse Relations and Discourse Marking* (cf. p. 65).

BURSTEIN, Jill, Daniel MARCU et Kevin KNIGHT (2003). « Finding the WRITE stuff : automatic identification of discourse structure in student essays ». In : *IEEE Intelligent Systems : Special Issue on Advances in Natural Language Processing* 18 (cf. p. 53, 65).

BUSQUETS, Joan, Laure VIEU et Nicholas ASHER (2001). « La SDRT : une approche de la cohérence du discours dans la tradition de la sémantique dynamique ». In : *Verbum* 23, p. 73–101 (cf. p. 19–22, 41).

CANDITO, Marie, Joakim NIVRE, Pascal DENIS et Enrique H. ANGUIANO (2010). « Benchmarking of statistical dependency parsers for French ». In : *Proceedings of ICCL* (cf. p. 129).

CARLSON, Lynn et Daniel MARCU (2001). *Discourse Tagging Reference Manual*. Rapp. tech. University of Southern California Information Sciences Institute (cf. p. 29, 36, 50, 56).

CARLSON, Lynn, Daniel MARCU et Mary Ellen OKUROWSKI (2001). « Building a discourse-tagged corpus in the framework of rhetorical structure theory ». In : *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue* (cf. p. 15, 25, 56).

– (2003). « Building a discourse-tagged corpus in the framework of rhetorical structure theory ». In : *Current Directions in Discourse and Dialogue*, p. 85–112 (cf. p. 30, 32, 44).

CARTONI, Bruno, Sandrine ZUFFEREY et Thomas MEYER (2013). « Annotating the meaning of discourse connectives by looking at their translation : The translation-spotting technique ». In : *Dialogue & Discourse* 4, p. 65–86 (cf. p. 65).

CHAI, Joyce Y. et Rong JIN (2004). « Discourse Structure for Context Question Answering ». In : *HLT-NAACL 2004 : Workshop on Pragmatics of Question Answering* (cf. p. 53, 65).

CHAN, Yee S. et Hwee T. NG (2005). « Word Sense Disambiguation with Distribution Estimation ». In : *Proceedings of IJCAI05* (cf. p. 124).

CHAWLA, Nitesh V., Kevin W. BOWYER, Lawrence O. HALL et W. Philip KEGELMEYER (2002). « SMOTE : Synthetic Minority Over-sampling Technique ». In : *Journal of Artificial Intelligence Research* (cf. p. 200).

CHELBA, Ciprian et Alex ACERO (2004). « Adaptation of maximum entropy capitalizer : Little data can help a lot ». In : *Computer Speech & Language* 20, p. 382–399 (cf. p. 124, 128).

CHEN, Yanqing, Bryan PEROZZI, Rami AL-RFOU et Steven SKIENA (2013). « The Expressive Power of Word Embeddings ». In : *Proceedings of ICML Workshop on Deep Learning for Audio, Speech, and Language Processing* (cf. p. 7, 165).

COLINET, Margot, Laurence DANLOS, Mathilde DARGNAT et Grégoire WINTERSTEIN (2014). « Uses of the prepostion « pour » introducing an infinitival clause : description, formal criteria and corpus annotation ». In : *4ème Congrès Mondial de Linguistique Française* (cf. p. 40).

COLLOBERT, Ronan et Jason WESTON (2008). « A Unified Architecture for Natural Language Processing : Deep Neural Networks with Multitask Learning ». In : *Proceedings of ICML* (cf. p. 7, 165, 199).

CONRATH, Juliette, Stergos AFANTENOS, Nicholas ASHER et Philippe MULLER (2014). « Unsupervised extraction of semantic relations using discourse cues ». In : *Proceedings of Coling* (cf. p. 41, 183, 188, 199).

CORMINBOEUF, Gilles (2013). « Une composante « émotive » dans les constructions articulées par un *et* d'opposition ? » In : *Journal of French Language Studies* (cf. p. 111).

CRAMMER, Koby, Ofer DEKEL, Joseph KESHET, Shai SHALEV-SHWARTZ et Yoram SINGER (2006). « Online Passive-Aggressive Algorithms ». In : *The Journal of Machine Learning Research* 7, p. 551–585 (cf. p. 72).

CRAVEN, Mark et Johan KUMLIEN (1999). « Constructing biological knowledge bases by extracting information from text sources ». In : *Proceedings of the International Conference on Intelligent Systems for Molecular Biology* (cf. p. 4).

CRISTEA, Dan, Nancy IDE, Daniel MARCU et Valentin TABLAN (1999). « Discourse structure and co-reference : An empirical study ». In : *Proceedings of ACL* (cf. p. 65).

CUNHA, Iria da, Juan Manuel Torres MORENO et Gerardo SIERRA (2011). « On the Development of the RST Spanish Treebank ». In : *Linguistic Annotation Workshop* (cf. p. 25).

DAELEMANS, Walter, Antal VAN DEN BOSCH et Jakub ZAVREL (1999). « Forgetting exceptions is harmful in language learning ». In : *Machine Learning* 34, p. 11–43 (cf. p. 126, 197).

DAI, Wenyuan, Qiang YANG, Gui-Rong XUE et Yong YU (2007). « Boosting for Transfer Learning ». In : *Proceedings of ICML* (cf. p. 126, 158, 197).

DANLOS, Laurence (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, et Processing, 343—370 (cf. p. 50, 64).

– (2006). « Capacité générative forte de RST, SDRT et des DAG de dépendances pour le discours ». In : *Traitement automatique des langues* 47(2), 169—198 (cf. p. 22, 26).

– (2009). « D-STAG : un formalisme d'analyse automatique de discours basé sur les TAG synchrones ». In : *Revue TAL* 50, p. 111–143 (cf. p. 24, 38, 39).

DANLOS, Laurence et Charlotte ROZE (2011). « Traduction (automatique) des connecteurs de discours ». In : *Proceedings of TALN* (cf. p. 65).

DANLOS, Laurence, Diégo ANTOLINOS-BASSO, Chloé BRAUD et Charlotte ROZE (2012). « Vers le FDTB : French Discourse Tree Bank ». In : *Proceedings of TALN* (cf. p. 27, 34, 47).

DANLOS, Laurence, Aleksandre MASKHARASHVILI et Sylvain POGODALLA (2015). « Grammaires phrastiques et discursives fondées sur les TAG : une approche de D-STAG avec les ACG ». In : *Proceedings of TALN* (cf. p. 24).

DAUMÉ III, Hal (2007). « Frustratingly Easy Domain Adaptation ». In : *Proceedings of ACL* (cf. p. 124, 126–128, 137).

DAUMÉ III, Hal et Daniel MARCU (2006). « Domain adaptation for statistical classifiers ». In : *Journal of Artificial Intelligence Research* 26, 101—126 (cf. p. 124, 128, 138, 144).

DAUMÉ III, Hal et Daniel MARCU (2009). « A Noisy-Channel Model for Document Compression ». In : *Proceedings of ACL* (cf. p. 53, 65).

DAUMÉ III, Hal, Abhishek KUMAR et Avishek SAHA (2010). « Frustratingly Easy Semi-supervised Domain Adaptation ». In : *Proceedings of the Workshop on Domain Adaptation for Natural Language Processing* (cf. p. 124).

DEGAND, Liesbeth et Anne-Catherine SIMON (2005). « Minimal Discourse Units : Can we define them, and why should we ? » In : *Proceedings of SEM-05. Connectors, discourse framing and discourse structure : from corpus-based and experimental analyses to discourse theories* (cf. p. 15).

DEKEL, Ofer (2009). « Distribution-Calibrated Hierarchical Classification ». In : *Proceedings of NIPS* (cf. p. 199).

DEKEL, Ofer, Joseph KESHET et Yoram SINGER (2004). « Large Margin Hierarchical Classification ». In : *Proceedings of ICML* (cf. p. 199).

DENIS, Pascal et Benoît SAGOT (2009). « Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art POS tagging with less human effort ». In : *Proceedings of PACLIC* (cf. p. 129).

DIETTERICH, Thomas G. (2000). « Ensemble Methods in Machine Learning ». In : *Proceedings of the International Workshop on Multiple Classifier Systems* (cf. p. 197).

DIETTERICH, Thomas G. et Ghulum BAKIRI (1995). « Solving Multiclass Learning Problems via Error-correcting Output Codes ». In : *Journal of Artificial Intelligence Research* 2.1, p. 263–286 (cf. p. 72).

ELWELL, Robert et Jason BALDRIDGE (2008). « Discourse Connective Argument Identification with Connective Specific Rankers ». In : *Proceedings of ICSC* (cf. p. 62).

FELLBAUM, Christiane (1998). *WordNet : An Electronic Lexical Database*. Bradford Books (cf. p. 199).

FENG, Vanessa Wei et Graeme HIRST (2012). « Text-level Discourse Parsing with Rich Linguistic Features ». In : *ACL (1)* (cf. p. 58).

– (2014). « A Linear-Time Bottom-Up Discourse Parser with Constraints and Post-Editing ». In : *Proceedings of ACL* (cf. p. 2, 58, 59).

FINKEL, Jenny Rose et Christopher D. MANNING (2009). « Hierarchical Bayesian Domain Adaptation ». In : *Proceedings of ACL-HLT* (cf. p. 124, 127, 128, 197).

FISHER, Seeger et Brian ROARK (2007). « The utility of parse-derived features for automatic discourse segmentation ». In : *Proceedings ACL* (cf. p. 56).

FLORIAN, Radu, Hany HASSAN, Abraham ITTYCHERIAH et al. (2004). « A statistical model for multilingual entity detection and tracking ». In : *In Proceedings of NAACL/HLT* (cf. p. 127).

FORBES-RILEY, Katherine et Bonnie WEBBER (2006). « Computing discourse semantics : The predicate-argument semantics of discourse connectives in D-LTAG ». In : *Journal of Semantics* 23, p. 55–106 (cf. p. 23, 24, 40, 50).

FREUND, Yoav et Robert E. SCHAPIRE (1997). « A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting ». In : *Journal of Computer and System Sciences* 55.1, p. 119–139 (cf. p. 83, 158).

GASTEL, Anna, Sabrina SCHULZE, Yannick VERSLEY et Erhard HINRICHS (2011). « Annotation of explicit and implicit discourse relations in the TüBa-D/Z treebank ». In : *Proceedings of GSCL* (cf. p. 26, 27, 67).

GHOSH, Sucheta, Richard JOHANSSON, Giuseppe RICCARDI et Sara TONELLI (2011). « Shallow Discourse Parsing with Conditional Random Fields ». In : *Proceedings of 5th International Joint Conference on Natural Language Processing* (cf. p. 62).

GHOSH, Sucheta, Giuseppe RICCARDI et Richard JOHANSSON (2012). « Global Features for Shallow Discourse Parsing ». In : *Proceedings of SIGDIAL* (cf. p. 62).

GILLICK, Dan (2009). « Sentence Boundary Detection and the Problem with the U.S. » In : *Proceedings of NAACL* (cf. p. 55).

GREFENSTETTE, Edward et Mehrnoosh SADRZADEH (2011). « Experimental Support for a Categorical Compositional Distributional Model of Meaning ». In : *Proceedings of EMNLP* (cf. p. 168).

GUO, Honglei, Huijia ZHU, Zhili GUO et al. (2009). « Domain Adaptation with Latent Semantic Association for Named Entity Recognition ». In : *Proceedings of NAACL-HLT* (cf. p. 127).

HARRIS, Zellig S. (1954). « Distributional structure ». In : *Word* 10(23), p. 146–162 (cf. p. 183).

HASTIE, Trevor, Robert TIBSHIRANI et Jerome FRIEDMAN (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc. (cf. p. 4).

HE, Haibo et Edwardo A. GARCIA (2009). « Learning from Imbalanced Data ». In : *IEEE Transactions on Knowledge and Data Engineering* 21, p. 1263–1284 (cf. p. 73, 74, 76, 200).

HE, Haibo et Yunqian MA (2013). *Imbalanced Learning : Foundations, Algorithms, and Applications*. Wiley-IEEE Press (cf. p. 73).

HERNAULT, Hugo, Helmut PRENDINGER, David A. DUVERLE et Mitsuru ISHIZUKA (2010). « HILDA : A Discourse Parser Using Support Vector Machine Classification ». In : *Dialogue and Discourse* 1, p. 1–33 (cf. p. 56–60, 107).

HIGGINS, Derrick, Jill BURSTEIN, Daniel MARCU et Claudia GENTILE (2004). « Evaluating Multiple Aspects of Coherence in Student Essays ». In : *HLT-NAACL* (cf. p. 53, 65).

HIONG, Siaw Nyuk, Narayanan KULATHURAMAIYER et Jane LABADIN (2012). « Towards Structure-Based Paraphrase Detection Using Discourse Parser ». In : *Information Retrieval and Knowledge Management* 2, p. 96–103 (cf. p. 65).

HIRSCHBERG, Julia et Diane LITMAN (1987). « Now Let's Talk About Now : Identifying Cue Phrases Intonationally ». In : *Proceedings of ACL* (cf. p. 50).

HOBBS, Jerry R. (1985). *On the Coherence and Structure of Discourse*. Rapp. tech. Center for the Study of Language et Information, Stanford University (cf. p. 27).

HONG, Yu, Xiaopei ZHOU, Tingting CHE et al. (2012). « Cross-argument Inference for Implicit Discourse Relation Recognition ». In : *Proceedings of the ACM International Conference on Information and Knowledge Management* (cf. p. 77, 78, 87, 91).

HOVY, Eduard, Mitchell MARCUS, Martha PALMER, Lance RAMSHAW et Ralph WEISCHEDEL (2006). « OntoNotes : The 90% Solution ». In : *Proceedings of HLT-NAACL* (cf. p. 45, 200).

HUANG, Jiayuan, Arthur GRETTON, Karsten M. BORGWARDT, Bernhard SCHÖLKOPF et Alex J. SMOLA (2007). « Correcting Sample Selection Bias by Unlabeled Data ». In : *Proceedings of NIPS* (cf. p. 126).

HUNTER, Julie et Laurence DANLOS (2014). « Because we say so ». In : *Proceedings of EACL Workshop on Computational Approaches to Causality in Language* (cf. p. 30, 112).

IBN FAIZ, Syeed et Robert E. MERCER (2013). « Identifying Explicit Discourse Connectives in Text ». In : *Advances in Artificial Intelligence*. Sous la dir. d'Osmar R. ZAÏANE et Sandra ZILLES. T. 7884. Lecture Notes in Computer Science. Springer Berlin Heidelberg, p. 64–76 (cf. p. 61).

IRUSKIETA, Mikel, María J. ARANZABE, Arantza Diaz de ILARRAZA et al. (2013). « The RST Basque TreeBank : an online search interface to check rhetorical relations ». In : *Proceedings of the Workshop RST and Discourse Studies* (cf. p. 26).

IRUSKIETA, Mikel, Iria da CUNHA et Maite TABOADA (2015). « A qualitative comparison method for rhetorical structures : identifying different discourse structures in multilingual corpora ». In : *Language Resources and Evaluation* 49.2, p. 263–309 (cf. p. 26).

JAYEZ, Jacques et Corinne ROSSARI (1998). « Pragmatic connectives as predicates : the case of inferential connectives ». In : *Predicative forms in natural language and in lexical knowledge bases*. Kluwer, p. 285–319 (cf. p. 38, 39, 41).

JI, Yangfeng et Jacob EISENSTEIN (2014a). « One Vector is Not Enough : Entity-Augmented Distributional Semantics for Discourse Relations ». In : *CoRR* (cf. p. 2, 77–79, 84, 85, 95–97, 104, 105, 162, 177, 178, 188, 189).

– (2014b). « Representation Learning for Text-level Discourse Parsing ». In : *Proceedings of ACL* (cf. p. 2, 59, 86, 95).

JIANG, Jing (2008). *A Literature Survey on Domain Adaptation of Statistical Classifiers* (cf. p. 6, 119, 124).

JIANG, Jing et ChengXiang ZHAI (2007). « Instance Weighting for Domain Adaptation in NLP ». In : *Proceedings of ACL* (cf. p. 124).

JOHANNSEN, Anders et Anders SØGAARD (2013). « Disambiguating explicit discourse connectives without oracles ». In : *Proceedings of IJCNLP* (cf. p. 2, 61, 134).

JOHNSON, Christopher R., Charles J. FILLMORE, Miriam R.L. PETRUCK et al. (2002). *FrameNet : Theory and Practice* (cf. p. 199).

JOSHI, Aravind (1987). « An introduction to Tree Adjoining Grammar ». In : *Mathematics of Language*. Sous la dir. d'A. MANASTER-RAME. Amsterdam : John Ben- jamins, p. 87–114 (cf. p. 23).

JOSHI, Aravind, Laura KALLMEYER et Maribel ROMERO (2003). « Flexible Composition in LTAG : Quantifier Scope and Inverse Linking ». In : *Proceedings of the International Workshop on Compositional Semantics* (cf. p. 24).

JOTY, Shafiq et Alessandro MOSCHITTI (2014). « Discriminative Reranking of Discourse Parses Using Tree Kernels ». In : *Proceedings of EMNLP* (cf. p. 2, 58).

JOTY, Shafiq, Giuseppe CARENINI et Raymond T. NG (2015). « CODRA : A Novel Discriminative Framework for Rhetorical Analysis ». In : *Computational Linguistics* 41 :3 (cf. p. 58).

JOTY, Shafiq R., Giuseppe CARENINI et Raymond T. NG (2012). « A Novel Discriminative Framework for Sentence-Level Discourse Analysis ». In : *Proceedings of EMNLP* (cf. p. 56).

JOTY, Shafiq R., Giuseppe CARENINI, Raymond T. NG et Yashar MEHDAD (2013). « Combining Intra- and Multi-sentential Rhetorical Parsing for Document-level Discourse Analysis ». In : *Proceedings of ACL* (cf. p. 58, 59).

KAMP, Hans (1981). « Evénements, représentations discursives et référence temporelle ». In : *Langages* 64, p. 34–64 (cf. p. 19).

KAMP, Hans et Uwe REYLE (1993). *From Discourse to Logic : Introduction to Model-theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. T. 42. Studies in Linguistics and Philosophy. Kluwer (cf. p. 19).

KESKES, Iskander, Farah BENAMARA et Lamia HADRICH BELGUITH (2014). « Learning Explicit and Implicit Arabic Discourse Relations ». In : *Journal of King Saud University (JKSU-CIS), Special Issue on Arabic NLP : Current State and Future Challenges* (cf. p. 26).

KINGSBURY, Paul et Martha PALMER (2002). « From treebank to propbank ». In : *Proceedings of LREC* (cf. p. 45, 199).

KIPPER, Karin, Hoa Trang DANG et Martha PALMER (2000). « Class-based construction of a verb lexicon ». In : *Proceedings of AAAI* (cf. p. 200).

KNOTT, Alistair (1997). « A Data-Driven Methodology for Motivating a Set of Coherence Relations ». Thèse de doct. University of Edinburgh (cf. p. 40, 46, 50, 51).

KOEHN, Philipp et Josh SCHROEDER (2007). « Experiments in Domain Adaptation for Statistical Machine Translation ». In : *Proceedings of the Workshop on Statistical Machine Translation* (cf. p. 124).

KONG, Fang, Hwee Tou NG et Guodong ZHOU (2014). « A Constituent-Based Approach to Argument Labeling with Joint Inference in Discourse Parsing ». In : *Proceedings of EMNLP* (cf. p. 62, 64).

KOO, Terry, Xavier CARRERAS et Michael COLLINS (2008). « Simple Semi-supervised Dependency Parsing ». In : *Proceedings of ACL-HLT* (cf. p. 165).

KRISHNAPURAM, Balaji, Lawrence CARIN, Mário A. T. FIGUEIREDO et Alexander J. HARTEMINK (2005). « Sparse multinomial logistic regression : Fast algorithms and generalization bounds ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, p. 957–968 (cf. p. 72).

LABUTOV, Igor et Hod LIPSON (2013). « Re-embedding words ». In : *Proceedings of ACL* (cf. p. 8, 183, 199).

LAN, Man, Yu XU et Zhengyu NIU (2013). « Leveraging Synthetic Discourse Data via Multi-task Learning for Implicit Discourse Relation Recognition ». In : *Proceedings of ACL* (cf. p. 6, 76, 78, 87, 92, 96, 124, 132, 133).

LAPATA, Mirella et Alex LASCARIDES (2004). « Inferring Sentence-internal Temporal Relations ». In : *Proceedings of HLT-NAACL* (cf. p. 93).

LASCARIDES, Alex et Nicholas ASHER (1993). « Temporal Interpretation, Discourse Relations and Commonsense Entailment ». In : *Linguistics and Philosophy* 16 (cf. p. 14, 19, 21).

LE, Quoc V. et Tomas MIKOLOV (2014). « Distributed Representations of Sentences and Documents ». In : *Proceedings of ICML* (cf. p. 168).

LE THANH, Huong, Geetha ABEYSINGHE et Christian HUYCK (2004). « Generating Discourse Structures for Written Text ». In : *Proceedings of Coling* (cf. p. 56, 57).

LEBRET, Rémi et Ronan COLLOBERT (2014). « Word Emdeddings through Hellinger PCA ». In : *Proceedings of ACL* (cf. p. 7, 9, 166, 171, 188, 199).

LEVIN, Beth (1993). *English verb classes and alternations : a preliminary investigation*. University of Chicago Press (cf. p. 81).

LI, Jiwei, Rumeng LI et Eduard H. HOVY (2014a). « Recursive Deep Models for Discourse Parsing. » In : *Proceedings of EMNLP* (cf. p. 2, 59).

LI, Junyi Jessy et Ani NENKOVA (2014a). « Addressing Class Imbalance for Improved Recognition of Implicit Discourse Relations ». In : *Proceedings of SIGDIAL* (cf. p. 73, 74, 76–79, 86, 96, 102).

– (2014b). « Reducing Sparsity Improves the Recognition of Implicit Discourse Relations ». In : *Proceedings of SIGDIAL* (cf. p. 78, 79, 84, 96, 102, 145, 162, 172).

LI, Qi (2012). *Literature survey : Domain Adaptation Algorithms for Natural Language Processing*. Rapp. tech. City University of New York (cf. p. 6, 123, 124).

LI, Yancui, Wenhe FENG, Jing SUN, Fang KONG et Guodong ZHOU (2014b). « Building Chinese Discourse Corpus with Connective-driven Dependency Tree Structure ». In : *Proceedings of EMNLP* (cf. p. 27).

LIESBETH, Degand, Nathalie LEFÈVRE et Yves BESTGEN (1999). « The impact of connectives and anaphoric expressions on expository discourse comprehension ». In : *Document Design* 1, p. 39–51 (cf. p. 111).

LIN, Ziheng, Min-Yen KAN et Hwee Tou NG (2009). « Recognizing Implicit Discourse Relations in the Penn Discourse Treebank ». In : *Proceedings of EMNLP* (cf. p. 2, 7, 58, 64, 73, 76–82, 84, 86, 92, 95–97, 100, 101, 104–106, 108, 145, 163).

LIN, Ziheng, Hwee Tou NG et Min-Yen KAN (2010). *A PDTB-styled end-to-end discourse parser*. Rapp. tech. National University of Singapore (cf. p. 50, 60, 61, 63).

– (2011). « Automatically Evaluating Text Coherence Using Discourse Relations ». In : *Proceedings of ACL-HLT* (cf. p. 65).

– (2014). « A PDTB-styled end-to-end discourse parser ». In : *Natural Language Engineering* 20, p. 151–184 (cf. p. 2, 60–63, 107, 134–136, 200).

LONG, Mingsheng, Jianmin WANG, Guiguang DING, Dou SHEN et Qiang YANG (2012). « Transfer Learning with Graph Co-Regularization ». In : *Proceedings of AAAI* (cf. p. 124).

LONGO, Laurence (2013). « Vers de moteurs de recherche « intelligents » : un outil de détection automatique de thèmes ». Thèse de doct. Université de Strasbourg (cf. p. 13).

LOUIS, Annie, Aravind JOSHI et Ani NENKOVA (2010a). « Discourse Indicators for Content Selection in Summarization ». In : *Proceedings of SIGDIAL* (cf. p. 65).

LOUIS, Annie, Aravind JOSHI, Rashmi PRASAD et Ani NENKOVA (2010b). « Using entity features to classify implicit discourse relations ». In : *Proceedings of SIGDIAL* (cf. p. 7, 85).

MAAS, Andrew L., Raymond E. DALY, Peter T. PHAM et al. (2011). « Learning Word Vectors for Sentiment Analysis ». In : *Proceedings of ACL-HLT* (cf. p. 8, 183).

MANN, William C. et Sandra A. THOMPSON (1988). « Rhetorical Structure Theory : Toward a functional theory of text organization ». In : *Text* 8, p. 243–281 (cf. p. 15–18, 32, 36).

MARCU, Daniel (1997a). « From Discourse Structures to Text Summaries ». In : *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, p. 82–88 (cf. p. 65).

– (1997b). « The Rhetorical Parsing, Summarization, and Generation of Natural Language Texts ». Thèse de doct. University of Toronto (cf. p. 15, 16, 18, 19, 50, 65).

– (1999). « A Decision-Based Approach to Rhetorical Parsing ». In : *Proceedings of ACL* (cf. p. 53, 55, 57).

– (2000). « The Rhetorical Parsing of Unrestricted Texts : A Surface-based Approach ». In : *Computational Linguistics* (cf. p. 55, 56, 59).

MARCU, Daniel et Abdessamad ECHIHABI (2002). « An Unsupervised Approach to Recognizing Discourse Relations ». In : *Proceedings of ACL* (cf. p. 5–7, 50, 76, 80, 82, 83, 85, 87–89, 101, 108, 110, 117, 130, 132, 134, 139, 145, 163, 169, 196).

MARCU, Daniel, Estibaliz AMORRORTU et Magdalena ROMERA (1999). « Experiments in constructing a corpus of discourse trees ». In : *Proceedings of the ACL Workshop on Standards and Tools for Discourse Tagging* (cf. p. 25, 55–57).

MARCUS, Mitchell P., Beatrice SANTORINI et Mary Ann MARCINKIEWICZ (1993). « Building a large annotated corpus of english : The Penn Treebank ». In : *Computational Linguistics* 19(2), 313—330 (cf. p. 25, 45).

MASASHI, Sugiyama et Müller KLAUS-ROBERT (2005). « Input-dependent estimation of generalization error under covariate shift ». In : *Statistics & Risk Modeling* (cf. p. 125).

MATSUSHIMA, Shin, Nobuyuki SHIMIZU, Kazuhiro YOSHIDA, Takashi NINOMIYA et Hiroshi NAKAGAWA (2010). « Exact Passive-Aggressive Algorithm for Multiclass Classification Using Support Class ». In : *Proceedings of the SIAM International Conference on Data Mining* (cf. p. 72).

MCCLOSKY, David, Eugene CHARNIAK et Mark JOHNSON (2010). « Automatic Domain Adaptation for Parsing ». In : *Proceedings of NAACL-HLT* (cf. p. 124).

MCDONALD, Ryan et Fernando PEREIRA (2006). « Online learning of approximate dependency parsing algorithms ». In : *Proceedings of EACL* (cf. p. 60, 129).

MEYER, Thomas (2011). « Disambiguating Temporal-contrastive Discourse Connectives for Machine Translation ». In : *Proceedings of ACL* (cf. p. 65).

MILTSAKAKI, Eleni, Rashmi PRASAD, Aravind JOSHI et Bonnie WEBBER (2004). « The penn discourse treebank ». In : *Proceedings of LREC* (cf. p. 26, 31, 51, 89).

MINTZ, Mike, Steven BILLS, Rion SNOW et Dan JURAFSKY (2009). « Distant Supervision for Relation Extraction Without Labeled Data ». In : *Proceedings of ACL-IJCNLP* (cf. p. 4).

MITCHELL, Jeff et Mirella LAPATA (2010). « Composition in Distributional Models of Semantics ». In : *Cognitive Science* 34.8, p. 1388–1439 (cf. p. 168, 198).

MNIH, Andriy et Geoffrey HINTON (2007). « Three New Graphical Models for Statistical Language Modelling ». In : *Proceedings of ICML* (cf. p. 165).

MOESCHLER, Jacques (2002). « Connecteurs, encodage conceptuel et encodage procédural ». In : *Cahiers de Linguistique Française* 24, p. 265–292 (cf. p. 39).

MOHRI, Mehryar, Afshin ROSTAMIZADEH et Ameet TALWALKAR (2012). *Foundations of Machine Learning*. MIT Press (cf. p. 4, 72, 118).

MOORE, Johanna D. et Martha E. POLLACK (1992). « A Problem for RST : The Need for Multi-level Discourse Analysis ». In : *Computational Linguistic* 18.4, p. 537–544 (cf. p. 19).

MORENO-TORRES, Jose G., Troy RAEDER, Rocío ALAIZ-RODRÍGUEZ, Nitesh V. CHAWLA et Francisco HERRERA (2012). « A unifying view on dataset shift in classification ». In : *Pattern Recognition* 45, p. 521–530 (cf. p. 119, 120).

MULLER, Philippe, Stergos AFANTENOS, Pascal DENIS et Nicholas ASHER (2012a). « Constrained decoding for text-level discourse parsing ». In : *Proceedings of COLING* (cf. p. 2, 33, 59, 60, 107).

MULLER, Philippe, Marianne VERGEZ-COURET, Laurent PRÉVOT et al. (2012b). *Manuel d'annotation en relations de discours de projet* ANNODIS. Rapp. tech. Carnets de Grammaire - Rapports internes de CLLE-ERSS (cf. p. 29, 35, 36, 43, 50).

OZA, Umangi, Rashmi PRASAD, Sudheer KOLACHINA, Dipti Misra SHARMA et Aravind JOSHI (2009). « The Hindi Discourse Relation Bank ». In : *Proceedings of LAW* (cf. p. 2, 67).

PALMER, David D. et Marti A. HEARST (1997). « Adaptive Multilingual Sentence Boundary Disambiguation ». In : *Computational Linguistics* 23.2, p. 241–267 (cf. p. 55).

PAN, Sinno Jialin et Qiang YANG (2010). « A Survey on Transfer Learning ». In : *IEEE Trans. on Knowl. and Data Eng.* P. 1345–1359 (cf. p. 6, 92).

PARDO, Thiago A. S. et Eloize R. M. SENO (2005). « Rhetalho : Um corpus de referência anotado retoricamente ». In : *Proceedings of Encontro de Corpora* (cf. p. 26).

PARK, Joonsuk et Claire CARDIE (2012). « Improving Implicit Discourse Relation Recognition Through Feature Set Optimization ». In : *Proceedings of SIGDIAL Conference* (cf. p. 7, 77, 79, 84, 86, 90, 93, 95, 96, 100, 102, 108, 163, 199).

PEARSON, Karl (1901). « On Lines and Planes of Closest Fit to Systems of Points in Space ». In : *Philosophical Magazine* 2 (6), p. 559–572 (cf. p. 166, 184).

PEDREGOSA, F., G. VAROQUAUX, A. GRAMFORT et al. (2011). « Scikit-learn : Machine Learning in Python ». In : *Journal of Machine Learning Research* 12, p. 2825–2830 (cf. p. 100, 101, 145).

PÉRY-WOODLEY, Marie-Paule, Nicholas ASHER, Patrice ENJALBERT et al. (2009). « ANNODIS : une approche outillée de l'annotation de structures discusives ». In : *Proceedings of TALN* (cf. p. 26).

PERY-WOODLEY, Marie-Paule, Stergos AFANTENOS, Lydia-Mai HO-DAC et Nicholas ASHER (2011). « La ressource ANNODIS, un corpus enrichi d'annotations discursives ». In : *Traitement Automatique des Langues, Ressources Linguistiques Libres* 52.3, p. 71–101 (cf. p. 33).

PETASIS, Georgios (2011). « Unsupervised Domain Adaptation based on Text Relatedness ». In : *Proceedings of RANLP* (cf. p. 124).

PITLER, Emily et Ani NENKOVA (2009). « Using Syntax to Disambiguate Explicit Discourse Connectives in Text ». In : *Proceedings of the ACL-IJCNLP* (cf. p. 2, 50, 134, 135).

PITLER, Emily, Mridhula RAGHUPATHY, Hena MEHTA et al. (2008). « Easily Identifiable Discourse Relations ». In : *Proceedings of COLING (Posters)* (cf. p. 2, 50, 63, 82, 134, 136).

PITLER, Emily, Annie LOUIS et Ani NENKOVA (2009). « Automatic sense prediction for implicit discourse relations in text ». In : *Proceedings of ACL-IJCNLP* (cf. p. 6, 7, 61, 63, 64, 68, 73, 76–84, 86, 91, 93, 94, 96, 100, 102, 110, 163).

POLÁKOVÁ, Lucie, Jiří MÍROVSKÝ, Anna NEDOLUZHKO et al. (2013). « Introducing the Prague Discourse Treebank 1.0 ». In : *Proceedings of IJCNLP* (cf. p. 27).

POLANYI, Livia (1985). « A Theory of Discourse Structure and Discourse Coherence ». In : *Proceedings of the Meeting of the Chicago Linguistics Society* (cf. p. 22).

POLANYI, Livia et Annie ZAENEN (2006). « Contextual Valence Shifters ». In : *Computing Attitude and Affect in Text : Theory and Applications*. Sous la dir. de James G. SHANAHAN, Yan QU et Janyce WIEBE. T. 20. The Information Retrieval Series. Springer Netherlands, p. 1–10 (cf. p. 53, 65).

PRASAD, Rashmi, Eleni MILTSAKAKI, Aravind JOSHI et Bonnie WEBBER (2004). « Annotation and Data Mining of the Penn Discourse TreeBank ». In : *Proceedings of the ACL Workshop on Discourse Annotation* (cf. p. 31).

PRASAD, Rashmi, Aravind JOSHI, Nikhil DINESH et al. (2005). « The Penn Discourse TreeBank as a resource for natural language generation ». In : *Proceedings of the Corpus Linguistics Workshop on Using Corpora for Natural Language Generation*, p. 25–32 (cf. p. 50).

PRASAD, Rashmi, Eleni MILTSAKAKI, Nikhil DINESH, Alan LEE et Aravind JOSHI (2006). *The Penn Discourse TreeBank 1.0. Annotation Manual*. Rapp. tech. Institute for Research in Cognitive Science, University of Pennsylvania (cf. p. 31, 46).

PRASAD, Rashmi, Eleni MILTSAKAKI, Nikhil DINESH et al. (2007). *The Penn Discourse Treebank 2.0 Annotation Manual* (cf. p. 28, 46, 49, 51, 78).

PRASAD, Rashmi, Nikhil DINESH, Alan LEE et al. (2008a). « The penn discourse treebank 2.0 ». In : *Proceedings of LREC* (cf. p. 2, 3, 27, 40, 45–47, 62, 63, 89).

PRASAD, Rashmi, Samar HUSAIN, Dipti Misra SHARMA et Aravind JOSHI (2008b). « Towards an Annotated Corpus of Discourse Relations in Hindi ». In : *Proceedings of IJCNLP* (cf. p. 27).

PRASAD, Rashmi, Aravind JOSHI et Bonnie WEBBER (2010). « Exploiting Scope for Shallow Discourse Parsing ». In : *Proceedings of LREC* (cf. p. 62).

PRASAD, Rashmi, Susan MCROY, Nadya FRID, Aravind JOSHI et Hong YU (2011). « The biomedical discourse relation bank ». In : *BMC bioinformatics* (cf. p. 27).

PRASAD, Rashmi, Bonnie WEBBER et Aravind JOSHI (2014). « Reflections on the Penn Discourse TreeBank, Comparable Corpora and Complementary Annotation ». In : *Computational Linguistics* (cf. p. 27, 30, 37, 47, 48, 50).

PUSTEJOVSKY, James, Patrick HANKS, Roser SAURÍ et al. (2003a). « The TIMEBANK Corpus ». In : *Proceedings of Corpus Linguistics* (cf. p. 36, 200).

PUSTEJOVSKY, James, José CASTAÑO, Robert INGRIA et al. (2003b). « Timeml : Robust specification of event and temporal expressions in text ». In : *Proceedings of International Workshop on Computational Semantics (IWCS-5)* (cf. p. 14).

RALAIVOLA, Liva (2010). *Quelques contributions en apprentissage statistique : bruit, non-IID et noyaux*. Habilitation à diriger des recherches. Université de Provence (cf. p. 118).

REESE, Brian, Julie HUNTER, Nicholas ASHER, Pascal DENIS et Jason BALDRIDGE (2007). *Reference manual for the analysis and annotation of rhetorical structure (version 1.0)*. Rapp. tech. Discor, University of Texas, Austin (cf. p. 26).

RIFKIN, Ryan et Aldebaro KLAUTAU (2004). « In defense of one-vs-all classification ». In : *Journal of Machine Learning Research* 5, p. 101–141 (cf. p. 72).

ROZE, Charlotte (2009). « Base lexicale des connecteurs discursifs du français ». Mém.de mast. Université Paris Diderot (cf. p. 38, 39, 130).

– (2011). « Towards a Discourse Relation Algebra for Comparing Discourse Structures ». In : *Proceedings of CID 2011 - Constraints In Discourse* (cf. p. 42).

– (2013). « Vers une algèbre des relations de discours ». Thèse de doct. Université Paris-Diderot (cf. p. 2, 26, 34, 37, 39).

ROZE, Charlotte, Laurence DANLOS et Philippe MULLER (2012). « LEXCONN : a French Lexicon of Discourse Connectives ». In : *Discours, Multidisciplinary Perspectives on Signalling Text Organisation* 10 (cf. p. 38–40, 51).

RUTHERFORD, Attapol et Nianwen XUE (2014). « Discovering Implicit Discourse Relations Through Brown Cluster Pair Representation and Coreference Patterns ». In : *Proceedings of EACL* (cf. p. 2, 7–9, 73, 74, 77, 79, 84–86, 94–97, 100, 102, 162–166, 171–173, 177, 178, 188, 189, 193).

– (2015). « Improving the inference of implicit discourse relations via classifying explicit discourse connectives ». In : *Proceedings of NAACL-HLT* (cf. p. 2, 6, 73, 74, 76, 77, 84–87, 91, 92, 94–97, 100, 102–104, 111, 123, 132, 134, 150, 151, 158, 159, 163, 164, 177, 180, 181, 188–191, 193, 197).

SAGAE, Kenji (2009). « Analysis of Discourse Structure with Syntactic Dependencies and Data-Driven Shift-Reduce Parsing ». In : *Proceedings of IWPT 2009* (cf. p. 56, 57, 59).

SAHLGREN, Magnus (2006). « The word-space model : Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces ». Thèse de doct. Stockholm University (cf. p. 184).

SANDERS, Ted (2005). « Coherence, causality and cognitive complexity in discourse ». In : *Proceedings of the International Symposium on the Exploration and Modelling of Meaning* (cf. p. 112).

SCHABES, Yves (1990). « Mathematical and computational aspects of lexicalized grammars ». Thèse de doct. University of Pennsylvania (cf. p. 23).

SCHAPIRE, Robert E. et Yoram SINGER (2000). « BoosTexter : A Boosting-based System for Text Categorization ». In : *Machine Learning* 39, p. 135–168 (cf. p. 89).

SCHAUER, Holger (2000a). « From Elementary Discourse Units to Complex Ones ». In : *Proceedings of SIGdial* (cf. p. 15).

– (2000b). « Using coreferences for coherence relations ». In : *Proceedings of ACL* (cf. p. 15).

SCHAUER, Holger et Udo HAHN (2001). « Anaphoric Cues for Coherence Relations ». In : *Proceedings of RANLP* (cf. p. 67).

SCHEFFER, Tobias (1999). « Error Estimation and Model Selection ». Thèse de doct. Technischen Universitet Berlin, School of Computer Science (cf. p. 141).

SCHOURUP, Lawrence (1999). « Discourse Markers ». In : *Lingua* 107(3), p. 227–265 (cf. p. 111).

SHIMODAIRA, Hidetoshi (2000). « Improving predictive inference under covariate shift by weighting the log-likelihood function ». In : *Journal of Statistical Planning and Inference* (cf. p. 125).

SORIA, Claudia et Giacomo FERRARI (1998). « Lexical marking of discourse relations - some experimental findings ». In : *Proceedings of the ACL Workshop on Discourse Relations and Discourse Markers* (cf. p. 2, 66, 113).

SORICUT, Radu et Daniel MARCU (2003). « Sentence level discourse parsing using syntactic and lexical information ». In : *Proceedings of NAACL* (cf. p. 56–58).

SPORLEDER, Caroline (2008). « Lexical Models to Identify Unmarked Discourse Relations : Does WordNet help ? » In : *JLCL* 23, p. 20–33 (cf. p. 68, 85).

SPORLEDER, Caroline et Mirella LAPATA (2005). « Discourse chunking and its application to sentence compression ». In : *In Proceedings of HLT/EMNLP* (cf. p. 53, 56, 65).

SPORLEDER, Caroline et Alex LASCARIDES (2005a). « Exploiting Linguistic Cues to Classify Rhetorical Relations ». In : *Proceedings of RANLP-05* (cf. p. 87–89).

– (2005b). « Exploiting linguistic cues to classify rhetorical relations ». In : *Proceedings of RANLP* (cf. p. 139).

– (2008). « Using Automatically Labelled Examples to Classify Rhetorical Relations : An Assessment ». In : *Natural Language Engineering* 14, p. 369–416 (cf. p. 2, 6, 66, 87, 89, 110, 114, 116, 117, 123, 129–132, 134, 138, 139).

STEDE, Manfred (2004). « The Potsdam Commentary Corpus ». In : *Proceedings of the ACL Workshop on Discourse Annotation* (cf. p. 2, 26, 27, 67).

– (2011). *Discourse Processing*. Morgan & Claypool (cf. p. 56).

STEDE, Manfred et Arne NEUMANN (2014). « Potsdam Commentary Corpus 2.0 : Annotation for Discourse Research ». In : *Proceedings of LREC* (cf. p. 27).

STEDE, Manfred et Carla UMBACH (1998). « DiMLex : A lexicon of discourse markers for text generation and understanding ». In : *Proceedings of COLING* (cf. p. 40, 50, 67).

STEINLIN, Jacques, Margot COLINET et Laurence DANLOS (2015). « FDTB1 : Repérage des connecteurs de discours en corpus ». In : *Proceedings of TALN* (cf. p. 27, 40, 46, 197).

STONE, Philip J. et John KIRSH (1966). *The General Inquirer : A Computer Approach to Content Analysis*. MIT Press (cf. p. 81).

STORKEY, Amos J. (2009). « When training and test sets are different : characterizing learning transfer ». In : sous la dir. de J. CANDELA, M. SUGIYAMA, A. SCHWAIGHOFER et N. eds. LAWRENCE. T. Dataset Shift in Machine Learning. MIT Press, 3—28 (cf. p. 73).

STORKEY, Amos J. et Masashi SUGIYAMA (2007). « Mixture Regression for Covariate Shift ». In : *Proceedings of NIPS* (cf. p. 128).

SUBBA, Rajen et Barbara DI EUGENIO (2009). « An Effective Discourse Parser That Uses Rich Linguistic Information ». In : *Proceedings of ACL-HLT* (cf. p. 2, 58, 66).

SUGIYAMA, Masashi, Shinichi NAKAJIMA, Hisashi KASHIMA, Paul VON BÜNAU et Motoaki KAWANABE (2008). « Direct importance estimation with model selection and its application to covariate shift adaptation ». In : *Proceedings of NIPS* (cf. p. 126).

SØGAARD, Anders (2011). « Data point selection for cross-language adaptation of dependency parsers ». In : *Proceedings of ACL* (cf. p. 126).

– (2013). *Semi-supervised learning and domain adaptation in natural language processing*. Morgan & Claypool (cf. p. 6, 120, 124).

TABOADA, Maite (2006). « Discourse markers as signals (or not) of rhetorical relations ». In : *Journal of Pragmatics* 38, p. 567–592 (cf. p. 2, 38, 66, 67).

TABOADA, Maite et Debopam DAS (2013). « Annotation upon Annotation : Adding Signalling Information to a Corpus of Discourse Relations ». In : *Dialogue and Discourse* 4(2), p. 249–281 (cf. p. 40, 41).

TABOADA, Maite et William C. MANN (2006a). « Applications of Rhetorical Structure Theory ». In : *Discourse Studies* 8, p. 567–588 (cf. p. 17, 53, 64).

– (2006b). « Rhetorical Structure Theory : looking back and moving ahead ». In : *Discourse Studies* 8, p. 423–459 (cf. p. 15, 40).

TELLJOHANN, Heike, Erhard W. HINRICHS, Sandra KÜBLER, Heike ZINSMEISTER et Kathrin BECK (2009). *Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z)*. Rapp. tech. Universität Tübingen (cf. p. 27).

THIONE, Gian Lorenzo, Martin Van den BERG, Livia POLANYI et Chris CULY (2004). « Hybrid Text Summarization : Combining External Relevance Measures with Structural Analysis ». In : *Proceedings of the ACL Workshop Text Summarization Branches Out* (cf. p. 53, 65).

TURIAN, Joseph, Lev-Arie RATINOV et Yoshua BENGIO (2010). « Word Representations : A Simple and General Method for Semi-Supervised Learning ». In : *Proceedings of ACL* (cf. p. 7, 164, 165, 171, 174).

TURNEY, Peter D. et Patrick PANTEL (2010). « From frequency to meaning : Vector space models of semantics ». In : *Journal of Artificial Intelligence Research*, p. 141–188 (cf. p. 184, 186).

VAN ASCH, Vincent (2013). « Macro-and micro-averaged evaluation measures » (cf. p. 71).

VARMA, Sudhir et Richard SIMON (2006). « Bias in error estimation when using cross-validation for model selection ». In : *BMC bioinformatics* 7.1, p. 91 (cf. p. 141).

VENANT, Antoine, Nicholas ASHER, Philippe MULLER, Pascal DENIS et Stergos AFANTENOS (2013). « Expressivity and comparison of models of discourse structure ». In : *Special Interest Group on Discourse and Dialogue*. Metz, France (cf. p. 26, 200).

VERBERNE, Suzan (2007). « Discourse-based answering of why-questions ». In : *Traitement Automatique des Langues, special issue on "Discours et document : traitements automatiques"* 47(2), 21—41 (cf. p. 53, 65).

VERSLEY, Yannick (2010). « Discovery of Ambiguous and Unambiguous Discourse Connectives via Annotation Projection ». In : *Proceedings of the Workshop on the Annotation and Exploitation of Parallel Corpora* (cf. p. 40, 65, 198).

– (2011). « Towards finer-grained tagging of discourse connectives ». In : *Proceedings of the Workshop Beyound Semantics : Corpus-based Investigations of Pragmatic and Discourse Phenomena* (cf. p. 2, 63, 199).

– (2013). « Subgraph-based Classification of Explicit and Implicit Discourse Relations ». In : *Proceedings of IWCS* (cf. p. 67).

VLIET, Nynke van der et Gisela REDEKER (2011). « Complex Sentences as Leaky Units in Discourse Parsing ». In : *Proceedings of Constraints in Discourse* (cf. p. 58).

VOLL, Kimberly et Maite TABOADA (2007). « Not All Words Are Created Equal : Extracting Semantic Orientation as a Function of Adjective Relevance ». In : *AI 2007 : Advances in Artificial Intelligence*. Sous la dir. de Mehmet A. ORGUN et John THORNTON. T. 4830. Lecture Notes in Computer Science. Springer Berlin Heidelberg, p. 337–346 (cf. p. 53, 65).

WANG, WenTing, Jian SU et Chew Lim TAN (2010). « Kernel Based Discourse Relation Recognition with Temporal Ordering Information ». In : *Proceedings of ACL* (cf. p. 63, 78, 79, 84, 85).

WANG, Xun, Sujian LI, Jiwei LI et Wenjie LI (2012). « Implicit Discourse Relation Recognition by Selecting Typical Training Examples ». In : *Proceedings of COLING 2012 : Technical Papers* (cf. p. 6, 73, 77, 79, 87, 90, 96, 123, 126, 132, 163, 197).

WEBBER, Bonnie (2004). « D-LTAG : extending lexicalized TAG to discourse ». In : *Cognitive Science* 28, p. 751–779 (cf. p. 23, 38).

WEBBER, Bonnie, Matthew STONE, Aravind JOSHI et Alistair KNOTT (2003). « Anaphora and Discourse Structure ». In : *Computational Linguistic* (cf. p. 39).

WEBBER, Bonnie, Marcus EGG et Valia KORDONI (2010). « Discourse Structure and Language Technology ». In : *Natural Language Engineering* (cf. p. 55, 60, 64).

WEISS, Gary M. (2013). « Foundations of Imbalanced Learning ». In : *Imbalanced Learning : Foundations, Algorithms, and Applications*. Sous la dir. d'Haibo HE et Yunqian MA. Wiley-IEEE Press (cf. p. 73).

WELLNER, Ben et James PUSTEJOVSKY (2007). « Automatically Identifying the Arguments of Discourse Connectives ». In : *Proceedings of EMNLP-CoNLL* (cf. p. 61).

WHISSELL, Cynthia M. (1989). *The dictionary of affect in language* (cf. p. 85).

WIDLÖCHER, Antoine et Yann MATHET (2009). « La plate-forme Glozz : environnement d'annotation et d'exploration de corpus ». In : *Proceedings of TALN* (cf. p. 26).

WILSON, Deirdre et Dan SPERBER (1990). « Forme linguistique et pertinence ». In : *Cahiers de Linguistique Française* 11, 345—359 (cf. p. 115).

WILSON, Theresa, Janyce WIEBE et Paul HOFFMANN (2005). « Recognizing Contextual Polarity in Phrase-level Sentiment Analysis ». In : *Proceedings of HLT-EMNLP* (cf. p. 81).

WINTER, Joost de (2013). « Using the Student's t-test with extremely small sample sizes ». In : *Practical Assessment, Research & Evaluation* 18.10 (cf. p. 141).

WOLF, Florian et Edward GIBSON (2005). « Representing Discourse Coherence : A Corpus-Based Study ». In : *Computational Linguistics* 31(2), p. 249–288 (cf. p. 27).

XUE, Gui-Rong, Wenyuan DAI, Qiang YANG et Yong YU (2008). « Topic-bridged PLSA for Cross-domain Text Classification ». In : *Proceedings of SIGIR* (cf. p. 127).

XUE, Nianwen, Hwee Tou NG, Sameer PRADHAN et al. (2015). « The CoNLL-2015 shared task on shallow discourse parsing ». In : *Proceedings of CoNLL* (cf. p. 2).

YAROWSKY, David (1995). « Unsupervised Word Sense Disambiguation Rivaling Supervised Methods ». In : *Proceedings of ACL* (cf. p. 69, 90).

YU, Hsiang-Fu, Fang-Lan HUANG et Chih-Jen LIN (2011). « Dual Coordinate Descent Methods for Logistic Regression and Maximum Entropy Models ». In : *Machine Learning* (cf. p. 100).

ZADROZNY, Bianca (2004). « Learning and Evaluating Classifiers Under Sample Selection Bias ». In : *Proceedings of ICML* (cf. p. 125, 197).

ZEYREK, Deniz et Bonnie WEBBER (2008). « A Discourse Resource for Turkish : Annotating Discourse Connectives in the METU Corpus ». In : *Proceedings of IJCNLP* (cf. p. 27).

ZEYREK, Deniz, Deniz Umit TURAN, Cem BOZSAHIN et al. (2009). « Annotating Subordinators in the Turkish Discourse Bank ». In : *Proceedings of LAW* (cf. p. 27).

ZHOU, Zhi-Min, Yu XU, Zheng-Yu NIU et al. (2010). « Predicting Discourse Connectives for Implicit Discourse Relation Recognition ». In : *Proceedings of the International Conference on Computational Linguistics* (cf. p. 79, 87, 93, 96, 132).

ZHU, Xiaojin (2005). *Semi-Supervised Learning Literature Survey*. Rapp. tech. Computer Sciences, University of Wisconsin-Madison (cf. p. 69).

ZHU, Xiaojin et Zoubin GHAHRAMANI (2002). *Learning from Labeled and Unlabeled Data with Label Propagation*. Rapp. tech. Carnegie Mellon University (cf. p. 69).