HAL
open science

Compression automatique ou semi-automatique de textes par élagage des constituants effaçables : une approche interactive et indépendante des corpus

Mehdi Yousfi-Monod

To cite this version:

Mehdi Yousfi-Monod. Compression automatique ou semi-automatique de textes par élagage des constituants effaçables : une approche interactive et indépendante des corpus. Informatique [cs]. Université Montpellier II - Sciences et Techniques du Languedoc, 2007. Français. NNT : . tel-00185367v3

HAL Id: tel-00185367

https://theses.hal.science/tel-00185367v3

Submitted on 2 Jun 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Numéro d'identification :

Académie de Montpellier

Université Montpellier II
— Sciences et Techniques du Languedoc —

Thèse

présentée à l'Université des Sciences et Techniques du Languedoc
pour obtenir le diplôme de DOCTORAT

Spécialité	: **Informatique**
Formation Doctorale	: **Informatique**
École Doctorale	: **Information, Structures, Systèmes**

Compression automatique ou semi-automatique de textes par élagage des constituants effaçables : une approche interactive et indépendante des corpus

par

Mehdi Yousfi-Monod

Soutenue le 16 novembre 2007 devant le Jury composé de :

Jacques Chauché, Professeur, Université Montpellier 2, ... Président
Violaine Prince, Professeur, Université Montpellier 2, ... Directrice de thèse
Jacques Vergne, Professeur, Université de Caen, ... Rapporteur
Jean-Luc Minel, Ingénieur de recherche, Université Paris 10, ... Rapporteur
Juan Manuel Torres-Moreno, Maître de conférences, Université d'Avignon, ... Examinateur
Augusta Mela, Maître de conférences, Université Montpellier 3, ... Examinateur

Numéro d'identification :

Académie de Montpellier

Université Montpellier II
— Sciences et Techniques du Languedoc —

Thèse

présentée à l'Université des Sciences et Techniques du Languedoc
pour obtenir le diplôme de DOCTORAT

Spécialité	: **Informatique**
Formation Doctorale	: **Informatique**
École Doctorale	: **Information, Structures, Systèmes**

Compression automatique ou semi-automatique de textes par élagage des constituants effaçables : une approche interactive et indépendante des corpus

par

Mehdi Yousfi-Monod

Soutenue le 16 novembre 2007 devant le Jury composé de :

Jacques Chauché, Professeur, Université Montpellier 2, ... Président
Violaine Prince, Professeur, Université Montpellier 2, ... Directrice de thèse
Jacques Vergne, Professeur, Université de Caen, ... Rapporteur
Jean-Luc Minel, Ingénieur de recherche, Université Paris 10, ... Rapporteur
Juan Manuel Torres-Moreno, Maître de conférences, Université d'Avignon, ... Examinateur
Augusta Mela, Maître de conférences, Université Montpellier 3, ... Examinateur

Remerciements

Ma thèse étant maintenant achevée, il m'est alors possible de prendre du recul sur le travail effectué, ses difficultés, et les différents soutiens et contributions qui m'ont été apportés par de nombreuses personnes, sous de nombreuses formes. Cette partie du mémoire est dédiée ces personnes. J'ai choisi d'utiliser un ordre alphabétique pour citer les différentes personnes impliquées, car je ne tiens pas à essayer d'établir un ordre sur des critères personnels, j'ai apprécié chaque soutien d'une façon différente et c'est ce que je souhaite rendre ici, avec autant de justesse que possible, tout en espérant n'oublier personne (pas facile, ils sont nombreux).

Je remercie tout d'abord les membres de mon jury, **Jean-Luc Minel**, **Jacques Vergne**, **Augusta Mela**, **Juan Manuel Torres-Moreno**, **Jacques Chauché** et **Violaine Prince** pour leur implication directe dans mon travail de thèse. Je donne davantage de détail sur chacun d'entre eux par la suite.

Lylia Abrouk est une amie de ma promotion qui m'a soutenu tout le long de ma thèse, de manière quasi-quotidienne. Elle a su me redonner confiance dans mes nombreuses périodes de doutes, liées plus ou moins étroitement à la thèse, sur le plan professionnel comme sur le plan personnel. Lylia m'a aussi recadré et motivé lorsque je me relâchais ou m'égarais, particulièrement dans les deux dernières années. J'ai pu compter sur son soutien du début à la fin. Même dans mes périodes asociales où je m'isolais de toute interaction humaine, Lylia est venue d'elle-même me sortir de ces phases descendantes pour me remettre sur les rails. Je lui dois une partie considérable de la réussite de ma thèse. Merci Lylia.

Je remercie **Béatrice Arnulphy et Tanja Neuendorf** qui ont travaillé, au cours d'un stage, sur l'étude de l'importance des compléments et l'influence du genre textuel sur la compression de texte. J'ai apprécié leur sympathie et l'efficacité de leur travail.

Je remercie **Nicolas Béchet**, collègue thésard arrivé 3 ans après moi dans l'équipe et co-citadin[1], pour sa sympathie, nos discussions sur nos travaux, sur linux/ubuntu/beryl/..., pour notre compétition excitante sur *Jeux de Mots* (voir à Mathieu Lafourcade).

Je remercie **Jacques Chauché**, collègue de mon équipe et président de mon jury de thèse, avec qui j'ai travaillé étroitement au niveau de la mise en œuvre et de l'évaluation de mon travail. Jacques a toujours été très disponible, il a pu être présent au laboratoire à chaque fois que j'avais besoin de lui (et pourtant Bouzigues ce n'est pas juste à côté). Jacques a été très réactif à toute requête que je lui ai exprimée, il a su me proposer des solutions efficaces aux nombreux problèmes techniques que j'ai rencontrés. Notre collaboration fut très appréciable grâce à sa jovialité constante.

[1] Le hasard a voulu qu'on s'installe à peu près en même temps dans le même petit village pommé, Assas, pour être ensuite dans la même équipe de recherche.

Je remercie **Christophe Crespelle**, ami de ma promotion avec qui j'ai eu de nombreuses discussions farfelues et souvent hors-thèse mais très intéressantes et amusantes.

Je remercie **Rolland Ducourneau**, Professeur de l'UM2, pour avoir accepté de faire partie de mon comité de thèse et pour ses conseils éclairés sur l'orientation de mes travaux.

Je remercie **François Dupressoir** qui a proposé, au cours d'un stage, une méthode d'exploitation d'une ressource lexicale pour la détermination des éléments sous-catégorisés par le verbe, afin d'évaluer leur importance syntaxique. François a effectué un travail conséquent, de façon sérieuse et minutieuse.

Je remercie **Patricia Durand**, amie de longue date, pour son soutien chaleureux et sa bonne humeur déjantée et délirante qui ont parsemé mes années de thèse de moments agréables dans des ambiances gaies et colorées.

Je remercie **Luc Fabresse**, un ami de ma promotion avec qui j'ai passé de nombreux moments de détentes fort agréables durant ma thèse (vacances, mariage, paintball, LANs. . .). Son perfectionisme technique m'a souvent aidé, notamment en programmation Latex (merci pour le tableau «pro» de la présentation de thèse ;-). Merci aussi à Isabelle pour sa sympathie et sa gaieté festive.

Je remercie **Abdelkader Gouaïch**, Maître de Conférences de l'UM2, pour ses nombreux conseils éclairés prodigués avec un recul manifeste, nos discussions intéressantes sur mon sujet de thèse, et sa capacité à me remotiver dans les pires moments.

Je remercie **Simon Jaillet**, un ami qui a commencé sa thèse deux ans avant moi. Simon m'a conseillé de nombreuses fois sur la façon d'aborder une thèse, de gérer mon temps et ma quantité de travail, de me fixer des objectifs raisonables. Il a souvent essayé, avec un succès relatif, de me communiquer son immuable décontraction naturelle afin de tempérer mes excés de stress fréquents. Ses nombreuses aides techniques, stratégiques et morales m'ont réellement aidées au cours de ma thèse.

Je remercie **Alain Joubert**, un associé de notre équipe avec qui j'ai partagé notre bureau de temps en temps durant ma dernière année. Alain est une personne très à l'écoute, qui rend service sans hésiter, et ceci même lorsqu'il n'y a pas de demande explicite. Nous avons partagé de nombreuses pauses cafés agréables, dont les sujets de discussions ont vite tournés autour de *Jeux de Mots* (voir à Mathieu Lafourcade).

Je remercie **Alexandre Labadié**, collègue thésard arrivé 2 ans après moi dans l'équipe, pour sa sympathie, son esprit de contradiction source de nombreuses discussions intellectuellement stimulantes, ses goûts en animes et jeux de rôles que je partage fortement, sa volonté de défendre la LEM[2] alors qu'au fond de lui il n'y croit évidemment pas, hein ;-)

Je remercie **Mathieu Lafourcade**, un collègue d'équipe, avec qui j'ai partagé mon (son) bureau durant une année. J'ai beaucoup apprécié son humour, sa bonne humeur

[2]Loi de l'Emmerdement Maximum.

(presque constante), ses sujets de discussions très souvent orientés vers le ludique, et sa rigueur scientifique (un peu trop) poussée qui m'a forcé à prendre du recul sur mon travail bon nombre de fois afin de l'améliorer. J'ai adoré le concept de *Jeux de Mots*[3], un jeu Internet issu d'un projet de recherche de Mathieu et inspiré de *Google Image Labeler*, qui a pour but d'enrichir une base lexicale de relations variées entre mots. Ce jeu est à la fois très intéressant sur le plan ludique, mais aussi très fructueux sur le plan scientifique.

Je remercie ma sœur **Charlotte Leclerc** qui fut ma co-locataire durant cette dernière année. Charlotte a su tempérer mon rythme de travail par de nombreux moments de détente, souvent assez festifs. J'ai beaucoup apprécié son énergie communicative (sauf peut-être au réveil ;-), son enthousiasme, sa générosité et son caractère humain qui ont beaucoup contribué à rendre plus agréable mon travail et surtout mes loisirs à la maison. Je remercie également ses amis, je pense en particulier à **Camille, Raphaël, Filentre, Pierre-Antoine, Julien, Félix et Numa**, sur qui nous pouvions compter pour apporter gaité et animation dans nos soirées. Charlotte, avec l'aide de sa mère, m'ont préparé un pot de thèse de très grande qualité, au plus grand bonheur des convives. Merci à vous deux.

Je remercie **Marie-Paule Lefranc**, Professeur de l'UM2 travaillant à l'Institut de Génétique Humaine de Montpellier. J'ai travaillé dans son équipe en bio-informatique après ma licence, ma maîtrise et mon DEA, dans le cadre de stages, vaccations et CDD traitants essentiellement sur la création d'outils d'alignements de séquences nucléotidiques en immunologie. Marie-Paule m'a fait confiance dès le début malgré mon manque d'expérience (sortant d'une licence en informatique, j'étais un peu léger), elle est parvenue à me faire comprendre sa problématique en biologie alors que ce domaine m'était presque inconnu, elle m'a appris à travailler en équipe, à être très rigoureux dans mon travail, à valoriser mes productions et compétences. Travailler dans son équipe fut un réel plaisir et une expérience très enrichissante. Je remercie également mes ex-collègues d'équipe, dont **Véronique, Denys, Chantal, Nathalie, Géraldine, Delphine, Joumana, Séverine, Céline, Élodie et Quentin**.

Je remercie **Augusta Méla**, associée de notre équipe, Maître de Conférences de l'UM3, pour avoir accepté d'être examinateur de ma thèse, pour son implication dans ma partie théorique linguistique, pour m'avoir aidé à trouver un équilibre entre algorithmes computationnels et modèle linguistique.

Je remercie **Jean-Luc Minel** pour avoir accepté de rapporter mon mémoire de thèse, pour la qualité de ses remarques de fond sur mon travail.

Je remercie **Isabelle Mougenot** pour avoir accepté de faire partie de mon comité de thèse, pour sa très grande sympathie, pour les agréables parties de tennis que nous avons

[3] `http://www.lirmm.fr/jeuxdemots`

jouées ensemble.

Je remercie **Jocelyne Nanard**, responsable de la formation doctorale en informatique lors de ma thèse, qui a su me conseiller, me guider, plusieurs fois au cours de ma thèse, lorsque je me considérais dans une impasse. Jocelyne a toujours pris très à cœur son travail dans l'école doctorale, s'y employant à 100 %, avec beaucoup de minutie, de sérieux et de rigueur.

Je remercie **Nicole Olivet**, qui a tenu l'accueil du laboratoire durant mes années de thèses. Nicole a exercé son travail d'une manière exceptionnelle, dépassant largement le *minimum syndical*, en prodiguant aux membres du laboratoire une assistance efficace dans nombreuses nos démarches. Nicole est une personne très souriante, sympathique et avenante qui contribue réellement à l'amélioration de la qualité de vie dans notre laboratoire (dommage que tu partes bientôt Nicole).

Je remercie **Robin Passama**, collègue de bureau au début de ma thèse. J'ai apprécié sa grande sympathie et son humour. Robin m'a aidé pour la réalisation d'un poster, et bien qu'il fût dans un domaine bien différent du mien, il a su utiliser son recul pour me fournir des remarques pertinentes.

Je remercie **Violaine Prince**, ma directrice de thèse (et de stage de DEA), qui m'a offert un encadrement de qualité, en m'autorisant une grande liberté de pensée et d'action, tout en restant extrêmement disponible pour répondre à toutes mes requêtes. Elle a su être patiente malgré nos quelques incompréhensions passagères, elle a su trouver les bons mots pour me motiver à chaque fois que je doutais sur l'évolution de mon travail, elle a su m'orienter dans les grandes lignes tout en me laissant libre de suivre ma propre voie. Violaine m'a considéré comme un collègue, elle ne m'a imposé aucune charge extérieure à mes travaux de thèse (ce fut même le contraire, car elle m'a allégé plusieurs fois de certaines tâches qui peuvent incomber au thésard), elle m'a appris à m'épanouir en tant que chercheur scientifique autonome. Mon entourage m'a prouvé que le type et la qualité d'encadrement varie considérablement d'une thèse à une autre, et je me considère comme chanceux d'avoir pu travailler avec Violaine.

Je remercie **Mathieu Roche**, un collègue d'équipe avec qui j'ai eu de nombreuses discussions sur mon sujet de thèse et la rédaction, particulièrement la partie évaluation de mon approche. Mathieu a été très ouvert à mes difficultés et m'a apporté une aide généreuse à tous les problèmes que je lui ai exposés.

Je remercie **Didier Schwab**, un ami et collègue d'équipe qui s'est impliqué dans mon travail de thèse par une grande disponibilité, des conseils scientifiques éclairés, un soutien moral constant, des relectures multiples... J'ai aussi apprécié son humour (on n'était d'ailleurs souvent que lui et moi à comprendre nos blagues), nos nombreuses discussions agréables, scientifiques ou autres, et sa franchise sur la qualité scientifique qui a contribué à améliorer mon travail.

Je remercie **Juan Manuel Torres-Moreno** pour avoir accepté d'être examinateur de ma thèse, pour sa sympathie et ses questions pertinentes.

Je remercie **John Tranier**, un ami de ma promotion avec qui je me suis serré les coudes pour finir le mémoire de thèse. John a été très présent, surtout durant toute la partie la plus pénible (la rédaction). Nos discussions sur mon sujet m'ont été très bénéfiques, et notre entre-aide efficace. Je le remercie aussi pour tous les moments de détentes que nous avons passés ensembles et qui m'ont permis de m'évader un peu de mon travail (sorties, bouffes, MM2...).

Je remercie **Jacques Vergne** pour avoir accepté de rapporter mon mémoire de thèse, pour sa rigueur scientifique, et ses nombreuses remarques qui m'ont permis d'améliorer la qualité de mon mémoire.

Je remercie **Djamila Yousfi**, la femme de mon père, pour son attention particulière tout le long de ma thèse, pour les nombreux bons plats qu'elle m'a préparé, lesquels m'ont été très profitables en cette période où le temps me manquait et où l'envie de cuisiner équilibré avait tendance à s'échapper. Je la remercie aussi pour avoir participé à la préparation de mon pot de thèse, qui fut d'une très grande qualité.

De nombreux amis ont été présents au cours de ma thèse, en m'apportant un soutien d'une manière plus indirecte. Je les remercie, en espérant oublier personne : merci à **Samra, Sofiann, Karim, Julia, Mickaël, Catherine, Sandy, Caroline Phommaline, Chédy Raïssi, Marc Plantevit, Pierre Larmande, Henri Frederico Eberspacher, Yolande et Ehoud Ahronovitz, Léa Nadai, Aude et Nicolas Barthes, Baptiste Munier, Jean-François et Florence, Vincent et Mélanie, Marc et Jenna, Clément et Isabelle Jonquet, Jean Privat, Vivienne, Pierre Bienfait**[4]**, Samir, Luc Granier, Aurélien**.

Merci aussi à l'ensemble des personnes qui ont participé à l'évaluation de mon approche.

Enfin, je remercie mes parents, **Régine et Abdallah**, pour tout l'amour, le soutien, l'éducation et les valeurs qu'ils m'ont transmis. J'ai pu grâce à eux m'ouvrir l'esprit à de nombreux domaines différents et passionnants.

[4]Pour la correction de mon résumé anglais.

Je dédie cette thèse à mes parents, Régine et Abdallah.

Sommaire

Table des figures **1**

Liste des tableaux **3**

1 Introduction **7**

1.1 Problématique 8

1.2 Contributions de cette thèse 10

Un modèle computationnel compatible avec le résumé par compression syntaxique. 10

Un logiciel semi-automatique et automatique de compressions de phrases. 10

Un protocole d'évaluation adapté à la compression semi-automatique et automatique de phrases. 10

1.3 Organisation de la thèse 11

2 Le résumé automatique de textes : panorama du domaine et état-de-l'art **13**

2.1 Introduction 13

2.2 La diversité des résumés 15

2.2.1 La source 15

2.2.1.1 Le médium 15

2.2.1.2 Le nombre de documents sources 15

2.2.1.3 La langue 16

2.2.1.4 Le domaine 16

2.2.1.5 Le genre 16

2.2.2 La cible 17

2.2.2.1 Le thème 17

2.2.2.2 Les événements 18

2.2.2.3 Période temporelle ou spatiale 18

2.2.2.4 Le style . . . 18
2.2.2.5 La taille . . . 18
2.2.2.6 Les éléments saillants . . . 18
2.2.3 La granularité des segments textuels analysés . . . 19
2.2.4 Le type d'information analysée . . . 20
2.2.5 La profondeur d'analyse . . . 20
2.2.6 Le processus de production . . . 20
2.2.6.1 Le résumé par reformulation . . . 21
2.2.6.2 Le résumé par extraction . . . 22
2.3 Les informations extraites pour la production du résumé . . . 23
2.3.1 Analyse en surface . . . 23
2.3.1.1 Fréquence des termes . . . 24
2.3.1.2 Marqueurs lexicaux de segments textuels . . . 25
2.3.1.3 Position de segments textuels dans le texte . . . 25
2.3.1.4 Nature des constituants . . . 26
2.3.2 Analyse des entités nommées . . . 26
2.3.2.1 La similarité lexicale . . . 26
2.3.2.2 La similarité thématique . . . 27
2.3.2.3 La coréférence . . . 27
2.3.3 Analyse de la structure du texte et de la phrase . . . 27
2.3.3.1 Structure thématique . . . 29
2.3.3.2 Structure événementielle . . . 29
2.3.3.3 Structure rhétorique . . . 30
2.3.3.4 Structure des rôles thématiques des phrases . . . 31
2.3.3.5 Structure syntaxique des phrases . . . 32
2.4 Les modèles de résumé automatique des logiciels commerciaux . . . 34
2.5 Conclusion . . . 35

3 La compression de phrases par élagage de l'arbre syntagmatique 37
3.1 Introduction . . . 37
3.1.1 La compression de phrases . . . 39
3.1.2 La compression syntaxique de phrases . . . 40
3.1.3 Nos objectifs . . . 41
3.1.4 Notre compression syntaxique de phrase . . . 42
3.2 Une classification des éléments effaçables . . . 43
3.2.1 Le gouvernement syntaxique. . . . 44

3.2.1.1 Tête gouvernante et constituant gouverné. 44
3.2.2 La classification des éléments effaçables 47
3.2.2.1 Le spécifieur 48
3.2.2.2 Le complément 49
L'argument. 50
Le complément dans le $TLFi$. 50
Le complément dans le Bescherelle. 51
3.2.2.3 L'adjoint 52
3.2.2.4 L'adverbe et le pronom en tant que têtes syntaxiques . . . 52
L'adverbe en tant que tête. 52
Le pronom en tant que tête. 54
3.2.2.5 Les constituants gouvernés par I^0 54
Spécifieur et compléments de I^0. 54
Adjoints de I^0. 55
3.2.2.6 Le modifieur 56
3.2.2.7 Notre classification 58
Le modifieur. 58
Le complément. 58
Les autres têtes lexicales. 58
Le système de règles structurelles adapté. 58
Récapitulatif. 59
Illustration de notre compression basée sur les modifieurs et compléments. 59
Conclusion. 61
3.3 Exploitation de traits linguistiques dans notre compression de phrases . . . 61
3.3.1 Exploitation de la sous-catégorisation 62
3.3.1.1 Des grammaires universelles aux ressources lexicales . . . 62
3.3.1.2 La sous-catégorisation dans le Le*fff* 64
3.3.1.3 Les compléments obligatoires dans le Le*fff* 66
Informations de sous-catégorisation partielles. 68
3.3.2 Les fonctions lexicales 69
3.3.3 Les autres traits linguistiques exploitables 73
3.3.3.1 Les phrasèmes complets 73
3.3.3.2 L'article 74
3.3.3.3 Les éléments incidents 75
3.3.3.4 Le modifieur du nom détaché 75

3.3.3.5 La position des constituants dans la phrase 76
3.3.3.6 La négation et l'interrogation 76
3.4 Esquisse d'un modèle computationnel 77
3.4.1 Nos objets linguistiques 78
3.4.2 Notre algorithme de compression syntaxique 79
3.5 L'influence du genre de texte sur l'importance des modifieurs et compléments 83
3.6 Les limites de la localisation du contenu important 84
3.7 Conclusion 85

4 Conception du compresseur de phrases 87
4.1 Introduction 87
4.2 Architecture 88
4.2.1 Première étape : analyse syntaxique 88
4.2.2 Seconde étape : sélection des constituants 90
4.2.3 Troisième étape : compression de phrases 90
4.3 Analyse syntaxique 91
4.3.1 SYGMART : un outil de manipulation d'éléments structurés 91
4.3.1.1 Caractéristiques du modèle d'analyse syntaxique de SYGFRAN 91
4.3.1.2 OPALE : le module de décomposition morphologique . . . 91
4.3.1.3 TELESI : le module de transformation d'éléments structurés 93
Exemple de rendu. 94
Le réseau de grammaires. 95
Exemple de grammaire TELESI. 95
4.3.1.4 AGATE : le module de linéarisation d'éléments structurés . 101
4.3.2 SYGFRAN : l'analyseur syntaxique 101
4.3.2.1 La grammaire de SYGFRAN 101
L'attachement des compléments. 101
L'attachement des modifieurs. 103
Les variables syntaxiques. 104
4.3.2.2 La couverture syntaxique de SYGFRAN. 104
Syntaxe ambiguë. 104
Analyse partielle. 105
4.3.2.3 Caractéristiques techniques de SYGFRAN 107
Modèle d'analyse syntaxique. 107
Volume d'informations. 108

Complexité de l'analyse. 109
Pourcentage de couverture syntaxique. 109
4.4 Le compresseur de phrases CoLin 110
4.4.1 Les règles de compression de CoLin 113
4.4.1.1 Grammaires de post-traitement à Sygmart 113
Choix arbitraires. 113
Corrections de l'analyse. 113
Construction syntaxique des verbes. 113
Gestion des formes contractées et composées. 115
4.4.1.2 Grammaires de résolution des anaphores 115
4.4.1.3 Grammaires de sélection des constituants 115
Définition des modifieurs et compléments. 116
Sélection des modifieurs et compléments. 116
Verrous sur l'effacement. 116
4.4.1.4 Grammaires de préparation à la linéarisation 117
Délimitation des constituants. 117
Encadrement des constituants. 118
Aplatissement de l'arbre. 119
4.4.1.5 Grammaire de linéarisation 121
4.4.2 Interface Web de CoLin 121
4.4.2.1 L'interaction dans le résumé automatique 122
4.4.2.2 L'interaction dans CoLin 123
La sélection des constituants. 124
Des couleurs pour l'importance. 124
L'inclusion des constituants. 125
Exemple de capture d'écran de l'interface de CoLin. 125
Le fonctionnement technique de l'interface Web de CoLin. . 127
4.5 Conclusion 128

5 Évaluation de notre approche sur la compression syntaxique des phrases 129

5.1 Introduction 129
5.2 L'évaluation de résumés 130
5.2.1 Évaluation automatique 131
5.2.1.1 ROUGE 132
5.2.2 Évaluation manuelle 133

5.3 Protocole d'évaluation de CoLin 134
5.3.1 Protocole d'évaluation de l'aide apportée par l'interaction dans CoLin 135
5.3.2 Protocole d'évaluation de la qualité des compressions produites par CoLin 136
5.3.2.1 Constitution d'un corpus de documents adéquat pour l'évaluation 137
La cohérence discursive. 137
Le genre des textes. 137
La taille des textes. 138
5.3.2.2 La notation des compressions 138
Les types de compression. 138
La présentation des compressions à noter. 139
La notation du contenu et de la cohérence. 140
5.3.3 Le système d'évaluation 141
5.3.3.1 Le système informatique 141
5.3.3.2 Les étapes de l'évaluation 142
Première étape : compression des documents. 142
Seconde étape : notation des paragraphes compressés. 144
Captures d'écran. 144
5.4 L'expérimentation 147
5.4.1 Le corpus de l'évaluation 147
5.4.1.1 Présentation 147
5.4.1.2 Prétraitement à l'évaluation 148
Intégration des cas syntaxiques des phrases du corpus. 148
Étiquetage morphologique du corpus. 149
Balises de sous-analyse syntaxique. 149
Règles transformationnelles *ad hoc*. 149
5.4.2 Résultats, discussion et bilan 150
5.4.2.1 Participation 150
Utilisateurs. 150
Compressions. 151
Notations. 151
Compressions et notations individuelles. 152
5.4.2.2 Résultats 152
Le temps de compression. 152
Satisfaction de l'interaction avec CoLin. 154

Le taux de compression. 154
La notation des compressions. 155
Nature des mauvaises notes. 157
5.4.2.3 Bilan 161
5.5 Conclusion 161

6 Conclusion et perspectives **165**
6.1 Synthèse 165
6.2 Perspectives 168
6.2.1 Traitement automatique de la sous-catégorisation 168
6.2.2 Apprentissage sur l'interaction 168
6.2.3 La compression de phrases dans le résumé automatique 169
Les marqueurs lexicaux des fonctions rhétoriques. 169
Le thème comme critère d'importance des constituants et phrases. 169

Index **171**

A Glossaire **173**
A.1 Sigles et acronymes 173
A.2 Catégories des têtes lexicales et fonctionnelles de l'approche théorique . . . 174
A.3 Variables de SYGFRAN et COLIN 174
A.4 Valeurs de SYGFRAN et COLIN 174

B Extraits des corpus exploités au cours de la thèse **177**
B.1 Conte polynésien correctement analysé par SYGFRAN 177
B.2 Extraits du corpus d'évaluation 179
B.2.1 Premier document narratif, *Vingt mille lieues sous les mers* de Jules Verne, extrait du premier chapitre 179
B.2.2 Premier document scientifique, extrait du corpus de la conférence DEFT'06 180
B.2.3 Premier document journalistique, extrait d'un article publié sur le site internet le 27 février 2007, intitulé « Darfour : la Cour pénale internationale désigne les criminels de guerre » 181

C Règles SYGMART **183**

Bibliographie **195**

Table des figures

2.1 Exemple de patron pour un événement de type catastrophe. 21

3.1 Exemple d'arbre syntagmatique selon la théorie X-barre. 46
3.2 Exemple de déterminant comme complément. 49
3.3 Exemple d'arbre syntagmatique incluant un constituant adverbial. 53
3.4 Exemple de groupe sujet et de groupe prédicat compléments de I^0. 55
3.5 Exemple d'adjoint de la proposition, placé à sa gauche. 57
3.6 Exemple d'adjoint de la proposition, placé à sa droite. 57
3.7 Exemple d'arbre syntagmatique selon nos règles structurelles (3.5–3.9). . . 60
3.8 Fusion d'un X' avec un XP. 80
3.9 Exemple de fusion des X' avec les XP. 80
3.10 Exemple d'indice de position des mots dans la phrase. 81

4.1 Architecture d'un système de compression syntaxique de phrase selon notre méthode de compression théorique. 89
4.2 Exemple simplifié de sortie du module OPALE de SYGFRAN. 92
4.3 Exemple de sortie du module TELESI pour SYGFRAN. 94
4.4 Arbre du schéma de reconnaissance de la règle `R_ATTACHE_GRIMPER` du code source 4.1 TELESI. 97
4.5 Exemple d'arbre identifié par le schéma de reconnaissance de l'arbre de la figure 4.4. 98
4.6 Arbre où l'attachement du complément du verbe de la figure 4.5 a été corrigé. 99
4.7 Exemple d'arbre identifié par la règle `R_ATTACHE_INDEPENDAMMENT` de la grammaire `ATTACHE_COMPLEMENT` du code source 4.1. 99
4.8 Arbre où l'attachement du complément de l'adverbe de la figure 4.7 a été corrigé. 100
4.9 Arbre où le sous-arbre du modifieur a été élagué selon la règle `R_SUPPRIME_MODIFIEURS` du code source 4.1. 100
4.10 Exemple de construction syntaxique ambiguë analysée par SYGFRAN. . . . 105
4.11 Exemple de catégorie lexicale déduite par SYGFRAN. 105

4.12 Exemple d'analyse partielle de SYGFRAN causée par une catégorie lexicale inconnue. 106
4.13 Exemple d'analyse partielle de SYGFRAN causée par une construction syntaxique inconnue. 106
4.14 Schéma de déploiement de COLIN. 112
4.15 Les réseaux de grammaires de COLIN. 114
4.16 Exemple d'arbre avant linéarisation. 118
4.17 Exemple d'arbre balisé avant linéarisation. 120
4.18 Exemple d'arbre aplati, avant linéarisation, avec M pour mot, C pour constituant et P pour ponctuation. 121
4.19 Exemple de chaîne de sortie du module OPALE de COLIN. 122
4.20 Capture d'écran de l'interface de COLIN. 126
4.21 Exemple de modification de la sélection par défaut. 126

5.1 Étapes de l'évaluation pour l'utilisateur et le système. 143
5.2 Page d'information sur la progression dans la tâche de compression. 145
5.3 Interface de compression manuelle (allégée d'une partie du texte). 145
5.4 Interface de compression semi-automatique (allégée d'une partie du texte). 146
5.5 Page d'information sur la progression dans la tâche de notation. 146
5.6 Extrait de l'interface de notation. 147
5.7 Temps de compression moyen. 154
5.8 Progression du temps de compression. 155
5.9 Notes de satisfaction sur l'interaction avec COLIN. 156
5.10 Taux de compression moyen, basé sur le comptage des mots. 157
5.11 Valeurs moyennes des notes par paragraphe. 159
5.12 Nombre et type des insatisfactions sur les mauvaises notes. 159

Liste des tableaux

2.1 5 solutions commerciales de résumé automatique. 35

3.1 L'effacement des modifieurs et compléments. 59
3.2 Le verbe *couper* dans le Le*fff*. 64
3.3 Exemple de compléments obligatoires dans la syntaxe du Le*fff*. 66
3.4 Les homonymes du verbe *voler* dans le Le*fff*. 67
3.5 Exemple de compléments obligatoires dans la syntaxe du Le*fff*. 68
3.6 Les fonctions lexicales standard pertinentes à notre approche. 72

4.1 Exemple d'informations renseignées par le module OPALE de SYGFRAN. . 93
4.2 Nom des variables et valeurs du code source 4.1. 97
4.3 Comparaison des quatre modèles d'analyse syntaxique présentés dans [Vergne, 2001] avec celui de SYGFRAN. 108
4.4 Volume d'informations dans SYGFRAN. 108
4.5 Valeurs des variables des balises de constituant de l'arbre de la figure 4.17. 119

5.1 Exemple de présentation de compressions à noter pour un document. . . . 140
5.2 Répartition du texte dans le corpus. 148
5.3 Participation à évaluation. 150
5.4 Répartition des nombres de textes compressés selon les genres. 151
5.5 Répartition des nombres de paragraphes notés selon les genres. 151
5.6 Répartition des nombres de textes compressés selon les évaluateurs. 152
5.7 Répartition des nombres de paragraphes notés selon les évaluateurs. 152
5.8 Temps de compression d'un document, en secondes. 153
5.9 Taux de compression. 156
5.10 Valeur moyenne des notes de paragraphe. 158
5.11 Nombre de paragraphes jugés insatisfaisants de type *préférence*. 158
5.12 Nombre de paragraphes jugés insatisfaisants de type *cohérence*. 160

Liste des codes source

4.1 Exemple de grammaire TELESI. 96
C.1 Extrait de la grammaire TELESI de COLIN de post-traitement à SYGFRAN pour la correction de constructions syntaxiques. 183
C.2 Extrait de la grammaire TELESI de COLIN de correction *ad hoc* des cas d'analyses partielles des phrases du corpus d'évaluation, ici pour le corpus journalistique. 184
C.3 Grammaire TELESI de COLIN de marquage des anaphores probables. . . . 185
C.4 Grammaire TELESI de COLIN de sélection des constituants. 188
C.5 Extrait de la grammaire TELESI de COLIN de préparation à la linéarisation finale. 191

1

Introduction

L'avènement du document numérique ainsi que la démocratisation des supports de stockage de masse et de l'échange de données à grande échelle ont multiplié, ces dernières décennies, la quantité des données textuelles accessibles par Internet comme dans les entreprises, administrations, *etc.* La manipulation de cet immense volume d'information ne peut être réalisée qu'en infime partie par un être humain. Par exemple, rechercher une information précise dans l'ensemble des pages accessibles librement sur Internet ne peut être réalisé manuellement, car ces pages se comptent maintenant en milliards. Le traitement automatique proposé par l'informatique voit alors une grande utilité dans ce type de tâches.

Lorsque les données sont structurées, comme les bases de données par exemple, l'humain parvient convenablement à les manipuler par un traitement automatique, car ces structures ont été crées dans ce but. Par contre, lorsqu'il s'agit de manipuler des documents textuels, rédigés dans une langue naturelle comme le français, les programmes informatiques voient de grandes difficultés à analyser ces données, qui ont été créées de manière naturelle par l'humain et pour l'humain. Ainsi les langues naturelles ne suivent pas de règles de structuration simples et rigides, et leur interprétation requiert un ensemble de connaissances fastidieuses et de procédés sophistiqués, adapté au fonctionnement et aux capacités de l'humain, mais pas à ceux de l'ordinateur.

L'humain, incapable de manipuler manuellement de grands volumes textuels, doit alors trouver des méthodes d'analyse automatique adaptées au traitement automatique de ces données. Ces méthodes s'inscrivent dans le domaine du traitement automatique des langues naturelles (TALN). Parmi les applications les plus connues, on peut citer la traduction automatique, le résumé automatique, la recherche d'information, la fouille de textes, la correction orthographique, la génération automatique de textes, la synthèse de la parole, la reconnaissance vocale ou la reconnaissance de l'écriture manuscrite.

Notre application est le résumé automatique de texte. Le principe général de la pro-

duction automatique de résumé est de trouver une fonction qui prend en argument un document textuel, et éventuellement quelques paramètres influençant le processus de production, et qui retourne un nouveau document textuel, plus petit que l'original et dont l'essentiel du contenu informationnel important est conservé.

Les premières approches dans ce domaine remontent à environ 50 ans et s'appuient sur des informations de surface, telles la fréquence et la répartition des mots, pour déterminer automatiquement l'importance des mots, phrases ou paragraphes. Dans [Luhn, 1958], l'une des toutes premières approches, les auteurs définissent un lien direct entre la fréquence d'un mot et son importance. Ils se basent sur l'hypothèse que l'auteur d'un texte utilise les mots clés du sujet développé plus fréquemment que les autres. Ils tiennent aussi compte des mots-outils[5] qui sont naturellement très fréquents dans tout texte, et généralement plus fréquents que les mots importants. Ainsi, ils définissent un intervalle de fréquence qui correspond aux mots généralement les plus importants, et s'appuient sur d'autres valeurs statistiques, comme la proximité de mots fréquents, pour définir l'importance des mots et phrases. Les résumés sont alors constitués par extraction puis concaténation des phrases les plus importantes.

Des approches de ce type ont ouvert la voie à l'exploration de techniques de production automatique de résumés. Ces dernières décennies, un très grand nombre d'approches ont abordé ce problème scientifique par une variété considérable de techniques et de ressources reposant sur des modèles mathématiques et parfois linguistiques. À l'heure actuelle, les meilleurs résumeurs automatiques sont encore loin de produire des résumés dignes de ceux produits par des être humains. Deux des principales difficultés sont (1) de restituer, dans un texte cohérent, les éléments dont le contenu informationnel est important, et (2) de conserver la structure du texte.

1.1 Problématique

Conserver la cohérence et la structure textuelle dans le résumé produit sont deux difficultés qui ne sont que superficiellement abordées dans la littérature. La majorité des approches se concentre sur la localisation du contenu informationnel important plutôt que sur la conservation de la cohérence dans le résumé produit. Les techniques utilisées tentent de localiser des paragraphes, phrases ou mots importants pour ensuite les extraire du document à résumer et enfin les mettre bout à bout pour produire le résumé final. Le résultat est assez souvent un agglomérat de segments textuels concaténés, d'une cohérence généralement faible et dont la structure est fortement dégradée à cause du procédé de

[5] Aussi appelés mots grammaticaux, ce sont les déterminants, les pronoms et les mots de liaison.

concaténation.

Exploiter une structure du texte peut permettre de remédier à ces deux problèmes. Plusieurs approches se sont orientées vers ce type de technique en exploitant principalement la structure rhétorique du texte ou sur la structure syntaxique de la phrase. La structure rhétorique est, à l'heure actuelle, extrêmement difficile à identifier, car elle relève en grande partie d'informations sémantiques, voire pragmatiques. La structure syntaxique de la phrase, bien que difficilement analysable de manière automatique, reste bien plus abordable que la précédente. De nombreux analyseurs syntaxiques existent et parviennent à identifier la catégorie grammaticale des mots, puis, de manière partielle, les relations fonctionnelles entre les mots ou syntagmes, permettant de dresser des arbres morpho-syntaxiques des phrases analysées. Dans le domaine du résumé automatique, cette structure peut être exploitée pour compresser les phrases, et ainsi produire des résumés par compression de texte.

Jean-Luc Minel, dans [Minel, 2007], relève l'intérêt et la problématique de la compression de phrases par ces mots :

> « Cela signifie aussi — et c'est l'un des inconvénients du système par extraction — que, si la phrase est très longue, elle sera mise telle quelle dans le résumé. Cela veut dire qu'on n'a pas trouvé des systèmes qui permettraient sur une phrase très longue de pouvoir retirer les informations redondantes, parce qu'on n'est pas capable à l'heure actuelle de dire ce qui est redondant dans une phrase. Par exemple, dire qu'une proposition relative est redondante, c'est un non-sens. Ainsi, si je dis « la fille qui a les yeux clairs » et si j'enlève le relatif « qui a les yeux clairs », il ne reste que « la fille » et c'est donc évidemment une information qui n'est plus pertinente. En terme de niveau de traitement, pour l'instant nous sommes donc essentiellement sur des traitements de type morphologique, avec un petit peu d'analyse morpho-syntaxique pour pouvoir distinguer si, par exemple, le terme « présente »[6] c'est le verbe conjugué, le nom ou l'adjectif. C'est à peu près les seuls niveaux qu'on utilise actuellement dans ces systèmes de traitement. »

La compression de phrases voit une utilité évidente pour les scientifiques, les journalistes et même les écrivains, qui sont souvent limités à une certaine taille de texte, et qui pour une raison ou une autre ont produit un texte origine plus long. Par ce procédé de résumé, il leur serait alors possible de réduire leurs textes de quelques pourcentages suffisants pour passer en dessous du seuil qui leur est imposé.

Les rares approches qui portent sur la compression syntaxique de phrases, comme [Knight & Marcu, 2002], le font à travers des modèles probabilistes basés sur des corpus

[6] 3 exemples

d'apprentissage. Ces approches sont contraintes par les limites intrinsèques d'un modèle et d'un corpus d'apprentissage. La généricité du corpus est inévitablement restreinte, notamment sur les distributions des constructions syntaxiques présentes et les genres textuels abordés. Ainsi, les systèmes informatiques issus de ces techniques se construisent sur les propriétés spécifiques de certains corpus, et donc ils en sont dépendants. Afin d'éviter ce type de dépendance, notre objectif est de développer une approche qui ne se calibre pas sur des corpus particuliers. Cela ne signifie pas que l'efficacité de l'approche sera constante quelque soit le corpus analysé. L'indépendance que nous visons se situe donc sur la définition de notre approche, et non pas sur la qualité des résumés produits par un système informatique basé sur notre approche.

1.2 Contributions de cette thèse

Un modèle computationnel compatible avec le résumé par compression syntaxique. Alors que les rares autres approches traitant de la compression de phrases le font à travers des modèles statistiques dont les performances sont très dépendantes des corpus d'apprentissages, nous proposons un modèle basé sur une étude linguistique de l'importance des constituants selon leur fonction syntaxique ainsi qu'un ensemble de traits linguistiques. Nous définissons comment procéder à une compression sans perte de cohérence syntaxique, et détaillons un ensemble de critères linguistiques qui renseignent sur l'importance des constituants de la phrase.

Un logiciel semi-automatique et automatique de compressions de phrases. À partir de ce modèle, nous avons développé COLIN, un logiciel de compression de textes. Ce dernier propose deux modes de résumé. Le mode semi-automatique propose une interaction inédite dans le domaine, où les constituants potentiellement effaçables sont mis en avant par un surlignage coloré selon leur importance probable, et où l'utilisateur peut modifier ce surlignage et ainsi la compression finale, par une simple interaction à base de clics de la souris. Le mode automatique génère une compression basée sur les critères d'importances définis dans notre modèle théorique. Les deux modes ont fourni d'excellents résultats lors de leur évaluation.

Un protocole d'évaluation adapté à la compression semi-automatique et automatique de phrases. Alors que les méthodes actuelles d'évaluation du résumé automatique proposent des techniques automatiques statistiques dépendantes de résumés de références subjectifs à leurs auteurs, ou des méthodes manuelles éprouvantes cognitivement pour les évaluateurs, nous proposons un protocole d'évaluation manuel où est demandé à l'évaluateur de réaliser des tâches naturelles pour un être humain, dans le but

de lui alléger son effort cognitif. Les résultats de l'évaluation ont confirmé le caractère subjectif des résumés produits par des être humains, et ainsi leur inadéquation à servir de résumé de référence pour une évaluation. De plus, notre protocole intègre l'évaluation du mode semi-automatique de notre compresseur, en comptabilisant 1) le temps gagné grâce à l'interface et 2) la satisfaction d'interaction.

1.3 Organisation de la thèse

Le chapitre 2 présente un panorama du domaine du résumé automatique et un état-de-l'art plus spécifique à notre approche. Dans une première partie nous décrivons la diversité des résumés, à travers un ensemble de facteurs qui peuvent influencer la production du résumé, du document source au document cible, sans rentrer dans le détail des approches existantes. Puis nous présentons un ensemble d'approches majeures du résumé automatique, organisées selon le type d'information extraite du document source. Enfin nous concluons en identifiant les pistes qui mènent à notre étude théorique.

Le chapitre 3 définit notre modèle computationnel de compression syntaxique. Après avoir introduit les bases de la compression ainsi que nos objectifs, nous définissons une classification des éléments syntaxiques effaçables à travers un ensemble de règles structurelles syntaxiques, en s'appuyant sur la théorie du gouvernement et du liage de Noam Chomsky. L'importance de ces éléments effaçables peut varier en fonction d'un ensemble de propriétés lexicales. Nous réalisons une étude de l'influence de ces propriétés. Nous synthétisons ensuite l'ensemble de notre étude théorique dans notre modèle computationnel de compression syntaxique. Enfin nous considérons l'influence du genre textuel sur notre compression et nous proposons de l'étudier davantage à travers une expérimentation.

Le chapitre 4 présente notre outil de compression de phrases CoLin. Nous commençons par présenter l'architecture globale de notre système de compression syntaxique de phrases, qui définit les étapes principales de la méthode de compression ainsi que les outils et ressources exploités. Ce système se veut autant que possible indépendant des différents choix des outils et de la mise en œuvre. Le reste du chapitre décrit nos choix à ce sujet, lesquels restent toutefois limités par les ressources linguistiques existantes et exploitables. Nous continuons en décrivant SYGFRAN, l'analyseur syntaxique utilisé dans notre système. Cet analyseur comprend un ensemble de règles créées manuellement, appliquées par un environnement opérationnel de transformation de structures arborescentes. Nous décrivons alors les caractéristiques globales SYGFRAN, qui ont motivé notre choix pour cet analyseur. Enfin, nous présentons CoLin, notre outil informatique de compression syntaxique de phrases, exploitant le résultat de l'analyseur syntaxique.

Le chapitre 5 aborde la validation de notre approche théorique par l'évaluation de

CoLin dans une expérimentation. Nous avons emprunté un chemin différent de la majorité des approches au sujet du protocole d'évaluation. Nous présentons la tendance classique dans le domaine puis nous définissons notre protocole et notre système d'évaluation. Nous terminons le chapitre par la description de l'expérimentation ainsi que l'étude et la discussion sur ses résultats.

Enfin, le chapitre 6 rappelle nos objectifs, notre travail réalisé et propose en perspectives un ensemble de pistes à explorer en continuité de notre travail.

2

Le résumé automatique de textes : panorama du domaine et état-de-l'art

Sommaire

2.1 Introduction 13
2.2 La diversité des résumés 15
2.3 Les informations extraites pour la production du résumé . . 23
2.4 Les modèles de résumé automatique des logiciels commerciaux 34
2.5 Conclusion 35

2.1 Introduction

Un résumé est, d'après le dictionnaire du Trésor de la Langue Française informatisé (TLFi), [TLF2007], une « présentation abrégée, orale ou écrite, qui rend compte de l'essentiel ». Cette définition est à la fois assez générale, car elle tient compte des résumés écrits et oraux, mais à la fois restreinte dans une certaine mesure, car elle ne prend pas en compte les résumés d'images, de vidéos ou de toute autre forme de communication. Notre travail se situant au niveau textuel, cette définition convient à notre approche. Dans ce chapitre, nous traiterons tout de même rapidement des autres types de médias avant de nous concentrer sur le médium concerné.

Afin de bien saisir cette définition du résumé, il est important de bien comprendre la notion d'*essentiel*. Elle peut se définir comme ce que le récepteur, c'est-à-dire la personne à qui est destiné le résumé, juge important. Or dans le domaine du traitement purement automatique de production à destination d'un récepteur, il importe de faire les choix de traitement indépendamment de ce dernier, car de tels systèmes de production ne disposent

pas d'information sur les préférences du récepteur au moment de la production. Seuls les systèmes faisant intervenir le récepteur dans le processus de production, c'est-à-dire les systèmes paramétrables ou semi-automatiques, peuvent tenir compte individuellement de chaque récepteur.

Notre approche théorique ne traitant pas de l'interaction d'un utilisateur[7] avec un système de production de résumé, nous ne discuterons pas de cet aspect à travers les approches présentées dans ce chapitre. Nous considérons alors la notion d'*essentiel* comme ce qu'un récepteur type juge important, c'est-à-dire ce qui est généralement jugé important. La tâche peut aussi influencer cette notion, spécialisant alors le récepteur type. Ainsi, les systèmes de résumé purement automatique tentent de localiser l'information importante pour un tel récepteur type, dans une certaine tâche.

Le domaine du résumé automatique est extrêmement vaste par son grand nombre d'approches et par leur diversité en techniques et ressources utilisées. Bien appréhender cet univers, différencier les approches et les catégoriser n'est pas une tâche évidente. Rares sont les travaux qui ont abouti à un état-de-l'art général et complet de ce domaine. L'une des approches se rapprochant au mieux de cet objectif est certainement [Alonso *et al.*, 2003b], dans laquelle plusieurs catégorisations des approches de résumé automatique sont proposées. Nous nous sommes appuyés sur cet article, en l'étoffant et l'orientant, pour construire notre état-de-l'art, ce qui nous a amené à notre propre catégorisation.

Le but de notre état-de-l'art est tout d'abord de présenter au lecteur un tour d'horizon des approches de résumé automatique, mais aussi de le plonger dans cet univers riche et hétérogène, en l'orientant progressivement vers notre approche, afin qu'il en saisisse bien le contexte et ces motivations.

Dans une première partie, section 2.2, nous décrivons la diversité des résumés, à travers un ensemble de facteurs qui peuvent influencer la production du résumé, du document source au document cible, sans rentrer dans le détail des approches existantes. Puis nous présentons un ensemble d'approches majeures au résumé automatique, organisées selon le type d'information extrait du document source, section 2.3. Nous poursuivons par un rapide aperçu des modèles de résumé automatique utilisés par quelques logiciels commerciaux, section 2.4. Enfin nous concluons en identifiant les pistes qui mènent à notre étude théorique, section 2.5.

[7] Nous explorerons cependant la piste de l'interaction dans les parties conceptuelle et expérimentale de ce travail.

2.2 La diversité des résumés

Un résumé est un terme général pouvant désigner un grand nombre de types de résumé. Pour caractériser ces différents types, nous avons choisi les critères suivants[8] : le type du document source (2.2.1), celui du document cible (2.2.2), mais aussi la granularité des segments textuels analysés (2.2.3), le type d'information analysée (2.2.4), la profondeur d'analyse (2.2.5) et le type de processus de production (2.2.6).

2.2.1 La source

Un premier critère de diversité est le type du document source, c'est-à-dire les caractéristiques qui le différencient d'un autre. Nous présentons ici ces principales caractéristiques habituellement prises en compte dans la production automatique d'un résumé.

2.2.1.1 Le médium

Le médium est le support du document. Nous comprenons document au sens d'un objet créé intentionnellement par une entité dans le but d'être interprété par une autre entité. Son support pourra être une image, un son, une vidéo, un texte, etc. Dans ce travail nous nous restreignons naturellement aux documents numérisés car ce sont eux que nous pourrons analyser automatiquement. En ce qui concerne la granularité, nous parlerons de document pour désigner un et un seul texte, même si, de manière générale, un document peut être un ensemble de textes ou une partie d'un texte.

Le sens commun de résumé concerne habituellement les documents écrits. Cependant il est possible de résumer d'autres média tels des enregistrements audio [Hori & Furui, 2001, Inoue *et al.*, 2004] ou vidéos (notamment pour les vidéos sportives : football [Ekin *et al.*, 2003], base-ball [Chang *et al.*, 2002], tennis [Coldefy *et al.*, 2004]), ou des images (comme [Carson *et al.*, 2002, Fei-Fei *et al.*, 2003] ou comme étape intermédiaire dans le processus de résumé vidéo). Le principe général reste le même : extraire du document une information jugée importante pour produire un document de plus petite taille. Dans notre approche et dans le reste de ce travail nous nous intéressons uniquement au résumé de textes.

2.2.1.2 Le nombre de documents sources

Résumer un seul document relève de la contraction de texte, alors que résumer plusieurs documents est de l'ordre de la synthèse de documents. Dans cette dernière, des difficultés supplémentaires interviennent comme la redondance informative, par exemple

[8] Ils correspondent aux principaux critères généralement abordés dans la littérature.

si on résume un ensemble de nouvelles traitant d'un même événement, ou comme le respect de la chronologie, car les informations extraites des différents documents doivent être insérées dans un ordre chronologique cohérent au niveau du résumé produit.

2.2.1.3 La langue

La plupart des informations utilisées pour produire le résumé (2.3) dépendent de la langue dans laquelle est rédigé le document source. Il est donc généralement nécessaire de se constituer un ensemble de ressources linguistiques pour chaque langue utilisée dans les documents à résumer. Ces ressources peuvent être constituées de marqueurs lexicaux (2.3.1.2), d'informations sur l'importance du positionnement des phrases ou paragraphes dans le texte (2.3.1.3), de fonctions lexicales (2.3.2.1), de concepts organisés sous la forme d'ontologies (2.3.2.2), de règles sur la grammaire de la langue (2.3.3.5), etc. Le résumé multi-langue devient alors possible en fonction de la disponibilité de ces ressources dans les langues choisies.

2.2.1.4 Le domaine

Le texte à résumer est inscrit dans un domaine, qu'il soit général ou spécialisé. Dans le cas de ce dernier, un vocabulaire spécialisé est généralement utilisé dans le texte. Pour les approches de résumé utilisant des techniques du niveau entité (2.3.2), il devient alors nécessaire de disposer des ressources lexicales correspondant à ce vocabulaire afin de reconnaître les entités associées, et de pouvoir restituer un vocabulaire adéquat.

2.2.1.5 Le genre

Le genre a une influence non négligeable dans le traitement automatique des textes. François Rastier écrit dans [Rastier, 2002] :

> « La typologie des genres paraît indispensable pour les traitements automatiques. Soit en général, car l'analyse des corpus en situation montre que le lexique, la morphosyntaxe, la manière dont se posent les problèmes sémantiques de l'ambiguïté et de l'implicite, tout cela varie avec les genres. Les systèmes d'analyse et de génération doivent tenir compte de ces spécificités, et les projets de systèmes universels sont ainsi irréalistes, linguistiquement parlant. Soit en particulier, car les genres sont déterminés par des pratiques sociales spécifiques, dans lesquelles les applications informatiques prennent place. Elles doivent donc tenir compte des contraintes propres aux pratiques où elles s'insèrent. »

Le résumé automatique ne faillit pas à cette règle, en effet de nombreuses approches se spécialisent sur un genre ou sous-genre particulier comme :

- les actualités : générales [Mani & Wilson, 2000, McKeown *et al.*, 1999], traitant de plusieurs événements [McKeown & Radev, 1995], [Daumé III *et al.*, 2002, Harabagiu & Lăcătuşu, 2002], traitant d'un seul événement [McKeown *et al.*, 2001, Daumé III *et al.*, 2002, Harabagiu & Lăcătuşu, 2002], traitant de catastrophes naturelles [Daumé III *et al.*, 2002, Harabagiu & Lăcătuşu, 2002, White *et al.*, 2001, Radev & McKeown, 1998]... ;
- les textes juridiques [Farzindar & Lapalme, 2005] ;
- les courriels [Somers *et al.*, 1997, Alonso *et al.*, 2003a] ;
- les textes biographiques [McKeown *et al.*, 2001, Daumé III *et al.*, 2002], [Harabagiu & Lăcătuşu, 2002] ;
- les articles scientifiques biomédicaux [Fiszman *et al.*, 2004], etc.

Ce phénomène se retrouve naturellement dans les conférences visant à évaluer les résumeurs automatiques comme la *Document Understanding Conference (DUC)*[9] qui propose des corpus de documents organisés par genre. Par exemple le genre proposé à DUC 2005 était les journaux d'actualité avec le *Financial Times of London* et le *Los Angeles Times*.

2.2.2 La cible

Une fois les différentes caractéristiques du document source prises en compte, les approches doivent considérer celles du document cible. Ce sont les différentes propriétés générales souhaitées pour le résumé produit. Le thème et la taille du résumé sont deux exemples de telles propriétés fréquemment utilisées dans les approches classiques au résumé automatique.

Nous présentons ces caractéristiques sous la forme d'un ensemble des critères de sélection du contenu informationnel. Nous traitons les plus utilisés et influents dans les approches de résumé automatique.

2.2.2.1 Le thème

Plusieurs thèmes (2.3.3.1) peuvent être présents au sein du document à résumer. Afin de gagner en contraction et d'affiner la qualité du résumé, il peut être utile d'orienter la conservation du contenu informationnel vers un thème choisi en accord avec le but du résumé. Ainsi un récepteur s'intéressant à la géologie peut préférer résumer une catastrophe naturelle aux aspects géologiques plutôt qu'aux aspects économiques ou humains.

[9] `http://duc.nist.gov/`

2.2.2.2 Les événements

Un document peut être constitué d'une succession et/ou imbrication d'événements (comme dans certains textes narratifs ou successions d'actualités) et le récepteur peut-être intéressé par un seul événement. Résumer un tel document pour un tel récepteur peut consister à conserver uniquement le contenu informationnel lié à cet événement.

2.2.2.3 Période temporelle ou spatiale

De manière similaire, le récepteur peut s'intéresser uniquement à un intervalle de temps ou à une zone spatiale. L'information non importante sera alors tout ce qui ne touche pas directement cette période.

2.2.2.4 Le style

Un résumé peut avoir plusieurs styles. Généralement il pourra être informatif, s'il couvre l'ensemble des sujets abordés dans le texte source ; indicatif, s'il propose un bref sommaire des principaux sujets adressés dans l'original ; agrégatif, s'il fournit de l'information non présente dans le texte source qui complète certaines de ses informations ou met en valeur des informations cachées ; ou enfin critique, s'il propose une valorisation additionnelle du texte résumé.

2.2.2.5 La taille

La taille du résumé influe grandement sur le contenu informationnel du résultat. Cette taille peut être déterminée par un pourcentage de la taille du texte original ou par une taille fixe. Dans le premier cas, les valeurs classiques s'échelonnent entre 1 % et 30 %, avec une valeur d'environ 10 % pour les résumés d'articles. Dans le cas des résumés multi-documents, la taille ne peut être déterminée à partir d'un pourcentage du texte source, une taille fixe sera alors préférée.

Selon les techniques de résumé utilisées dans les différentes approches du domaine, la configuration du paramètre de la taille peut être assez restreinte. Par exemple dans [Knight & Marcu, 2000], un des deux modèles utilise une technique déterministe et non paramétrable de compression de phrases, ce qui aboutit à une unique taille de compression pour un ensemble de phrases donné.

2.2.2.6 Les éléments saillants

La saillance linguistique met « en avant un élément du message, elle dirige l'attention du sujet sur cet élément et privilégie sa prise en compte dans le processus d'interprétation », écrit Frédéric Landragin dans [Landragin, 2004]. Un élément informatif est donc

saillant d'un point de vue linguistique s'il est mis en avant dans le document, que ce soit fait volontairement ou non par l'auteur. Pour produire un résumé selon cette information, plusieurs types de saillance linguistique peuvent être considérés.

[Landragin, 2004] distingue deux catégories générales de saillance, la première basée sur des facteurs physiques liés à la forme de l'énoncé, et la seconde sur des facteurs physiques liés au sens de l'énoncé. La première est découpée en six sous-catégories : la saillance

- intrinsèque au mot ;
- due à une mise en avant explicite lors de l'énonciation ;
- due à une construction syntaxique dédiée ;
- syntaxique liée à l'ordre et à la fréquence d'apparition des mots ;
- liée aux fonctions grammaticales ;
- indirecte par transfert grammatical de saillance.

La seconde se découpe aussi en six sous-catégories : la saillance

- liée à la sémantique des mots ;
- liée au rôle thématique ;
- liée au thème et au topique de l'énoncé ;
- liée au propos de la conversation ;
- liée à des inférences ;
- indirecte par transfert sémantique de saillance.

Selon le type de saillance considéré pour localiser l'information importante d'un document, plusieurs types de résumés automatiques peuvent être produits.

2.2.3 La granularité des segments textuels analysés

Lors du processus de production du résumé, l'analyse va porter sur différentes parties du ou des documents. Les différents niveaux de granularité comportent habituellement le texte complet, les sections et sous-sections, les paragraphes, les phrases, les propositions, les constituants[10], les items lexicaux[11] ou les mots. Ces segments sont généralement appelés unités textuelles (UT).

Les UT sont tout d'abord extraites du document source, puis certaines sont utilisées pour déterminer l'importance d'autres UT.

Par exemple, lorsque l'approche utilise une information globale IG à un ensemble de phrases, voire au texte, c'est généralement dans le but d'en extraire une donnée de type sémantique (comme le thème) et de la comparer avec celle d'unités textuelles de granularité

[10] Nous les définissons dans le prochain chapitre.

[11] Un item lexical est une suite de caractère formant une unité sémantique et pouvant constituer une entrée de dictionnaire.

plus fine afin d'estimer leur concordance avec IG et d'en déduire leur importance. Dans le cas du thème, des travaux comme [Harabagiu *et al.*, 2003, Lin & Hovy, 2000] génèrent (comme information IG) des signatures de thèmes.

Ensuite, les UT dont l'importance a été déterminée comme suffisamment élevée sont utilisées pour produire le résumé final.

Par exemple, si ces UT sont de l'ordre de la phrase, elles seront généralement mises bout à bout pour générer le résumé. Cette granularité est souvent utilisée car elle correspond à l'unité textuelle la plus petite qu'on puisse facilement supprimer sans toucher à la cohérence grammaticale. Par contre, résumer en plaçant bout à bout un ensemble de phrases extraites du document source peut poser au moins deux problèmes conséquents :

- la cohérence discursive est fortement réduite car les phrases supprimées contribuaient à la continuité discursive, même si elles étaient moins importantes ;
- certaines phrases longues et importantes (comme dans certains genres de textes narratifs) peuvent posséder des informations non indispensables à la compréhension globale. Ces approches n'entreront pas dans la phrase pour éliminer le contenu peu important.

Le premier problème nécessite la considération d'informations structurelles discursives (voir section 2.3.3), le second requiert naturellement un traitement à une granularité plus fine que la phrase.

2.2.4 Le type d'information analysée

Selon les approches, différents types d'informations sont extraits des textes, comme la fréquence des termes, leur nature et fonction, les relations entre les termes ou entités, le thème des segments textuels et la position des phrases ou paragraphes dans le texte. La section 2.3 y est entièrement consacrée.

2.2.5 La profondeur d'analyse

La nature du procédé d'extraction des informations analysées permet de définir la profondeur d'analyse. Cette dernière dépend donc du type d'information utilisée, nous le détaillons en section 2.3.

2.2.6 Le processus de production

La complexité de production automatique d'un résumé a conduit la majorité des approches à utiliser un processus de production dégradé par rapport à celui d'un être humain, mais grandement simplifié pour un traitement automatique. Produire un résumé au sens humain, c'est-à-dire effectuer à la fois un travail de contraction et de reformulation, n'est

Catastrophe	tornade
Coût des dégâts	100 millions de dollars
Nombre de morts	40
Lieu	Floride
Date	la semaine dernière

FIG. 2.1 – Exemple de patron pour un événement de type catastrophe.

à l'heure actuelle que très peu abordable de manière automatique. Quelques approches ont tout de même abordé ce sujet difficile (section 2.2.6.1). Les résumés produits de la sorte sont appelés *abstracts* dans le domaine. Les autres approchent préfèrent extraire des morceaux de texte du ou des document source pour composer le résumé final (section 2.2.6.2). On parle ici de production d'*extracts*.

2.2.6.1 Le résumé par reformulation

Nous appelons *résumé par reformulation* (ou *abstract*) un texte de taille plus petite que le document auquel il se réfère, et dont le sens se veut être le plus proche possible de celui du document, sans pour autant utiliser des phrases ou des portions du document initial. Les approches tentant d'aborder le résumé par un tel processus de production comme [McKeown & Radev, 1995, Radev & McKeown, 1998, Aone *et al.*, 1998, McKeown *et al.*, 1999, White *et al.*, 2001, Daumé III *et al.*, 2002] utilisent des structures de données intermédiaires avant la production du résumé, dans lesquelles sont extraites des informations du ou des document source telles les différents événements (section 2.2.6.1) ou les structures de type prédicat - arguments (section 2.2.6.1). Les structures de données une fois nourries par ces informations sont fournies en entrée à des outils de génération de langue (comme FUF/SURGE de [Elhadad & Robin, 1996]) qui génèrent des phrases grammaticalement correctes dans la langue désirée.

Les événements Les structures de données utilisées sont généralement des patrons [Radev & McKeown, 1998, White *et al.*, 2001]. Un patron correspond à un type d'événement particulier et contient un ensemble de champs, caractéristiques de l'événement correspondant, chacun nécessitant une valeur extraite du texte. Par exemple, un patron de type catastrophe est présenté en figure 2.1.

Ces valeurs sont généralement extraites par des techniques classiques d'extraction d'information, comme celles basées sur l'utilisation des marqueurs lexicaux [Mani & Wilson, 2000]. Une fois le ou les patrons remplis, une phrase peut être assez facilement générée à partir des champs. Pour l'exemple précédent, la phrase suivante pourrait être générée automatiquement : *La tornade de la semaine dernière qui a eu lieu*

en Floride a fait 40 morts et a coûté 100 millions de dollars.

Pour chaque type d'événement, un patron différent doit être manuellement créé. Le type des documents analysables par de telles approches restent alors limité par les types de patron créés.

Les structures prédicat - arguments Elles sont utilisées pour « représenter les actions par le biais de prédicats et les objets par les arguments des prédicats concernés. Les relations entre un prédicat et ses arguments sont exprimées par le biais de rôles thématiques qui sont assignés aux arguments du prédicat » [Pugeault, 1995].

Sans pousser jusqu'à l'extraction du rôle thématique, l'information de relation entre prédicat et argument peut être utilisée dans le résumé automatique.

Par exemple l'approche de [McKeown *et al.*, 1999] consiste à identifier les thèmes (2.3.3.1) principaux d'une série de documents traitant du même événement. Chaque thème est représenté par un ensemble de paragraphes, chacun provenant d'un document différent. Puis des phrases de ces paragraphes sont identifiés les différents prédicats et arguments. Les phrases partageant les mêmes prédicats et arguments vont alors être fusionnées, grâce à FUF/SURGE, moyennant un ajustement des informations pour convenir au format d'entrée de cet outil.

2.2.6.2 Le résumé par extraction

Les méthodes par extraction sont fondées sur l'hypothèse « qu'il existe, dans tout texte, des *unités textuelles saillantes* » [Minel, 2004]. Ces dernières représentent des points focaux, qui, soit expriment l'apport sémantique ou conceptuel du texte, soit permettent de le représenter dans sa globalité. Dès lors, le résumé par extraction cherche à repérer ces unités saillantes et propose un texte de taille plus petite que le document initial qui garde majoritairement ces unités. Nous faisons également l'hypothèse de l'existence de ces unités, ainsi que de leur intérêt pour le résumé. Le procédé de résumé par extraction consiste alors à localiser ces unités saillantes, puis à extraire certains segments textuels les incluant, pour ensuite composer le résumé final avec ces segments.

2.3 Les informations extraites pour la production du résumé, en fonction de la profondeur d'analyse

Dans cette partie, nous présentons les principaux types d'approches de résumé automatique, organisés par profondeur d'analyse. Par cette dernière nous entendons le niveau de complexité du traitement nécessaire pour extraire du document source les informations désirées. Les informations externes au document source[12] ne sont pas prises en compte dans la catégorisation. Cette catégorisation des résumeurs se base sur l'approche de [Alonso *et al.*, 2003b], qui décrit trois catégorisations différentes, dont la première sur laquelle nous nous sommes basé : analyse en surface (section 2.3.1), analyse des entités (section 2.3.2) et analyse du discours (section 2.3.3). Elle n'a pas pour objectif d'être exhaustive, seuls les principaux types d'informations extraites sont présentés ici. Ce n'est pas une catégorisation exclusive au niveau des approches, car certaines de ces dernières, et particulièrement les plus récentes, ne se restreignent pas à un seul niveau d'analyse, ni à une seule métrique mais préfèrent les combiner afin d'améliorer la qualité des résumés produits.

2.3.1 Analyse en surface

L'analyse en surface consiste à utiliser des informations qui peuvent être extraites du document par un traitement léger et direct telles la fréquence des termes (section 2.3.1.1), les marqueurs lexicaux (section 2.3.1.2), la position des segments textuels dans le texte (section 2.3.1.3) ou la nature des constituants (section 2.3.1.4). Ce type d'information, est par définition facile à extraire, il est utilisé dans les toutes premières approches de résumé automatique, comme [Luhn, 1958]. De nos jours, les informations de surface continuent à être largement exploitées, principalement dans des approches hybrides[13] comme [Radev *et al.*, 2004, Erkan & Radev, 2004]. La plupart des approches exploitant uniquement ce type d'information estiment l'importance de segments textuels indépendants les uns des autres et réalisent des résumés par extraction dont la cohérence structurelle reste assez faible (voir section 2.3.3). Cependant, cette information peut se révéler très utile comme critère supplémentaire dans la sélection des éléments de contenus importants ou non importants.

[12] Ce peut être un corpus de documents utilisé pour un apprentissage, des règles de grammaire pour une analyse syntaxique, une ontologie de concepts...

[13] mélangeant informations de surface avec informations au niveau entité ou structurel.

2.3.1.1 Fréquence des termes

De nombreuses approches, comme [Luhn, 1958, Barzilay & Elhadad, 1997], [Goldstein *et al.*, 2000, Boguraev & Neff, 2000, Lin & Hovy, 2002, Radev *et al.*, 2004], [Erkan & Radev, 2004], utilisent la fréquence des termes comme critère d'importance. La formule la plus utilisée est la propriété $tf \times idf$, définie par Salton [Salton & Yang, 1973], qui exprime qu'un terme est d'autant plus important qu'il est à la fois fréquent dans le document analysé et peu fréquent dans le corpus de documents analysé. Une fois les termes les plus importants déterminés, le résumé consiste généralement à conserver les phrases contenant le plus de ces termes. Cependant certaines approches travaillent à des niveaux de granularité inférieurs à la phrase.

Par exemple [Ishikawa *et al.*, 2002] utilise un catégoriseur SVM (*Support Vector Machine*) pour sélectionner les constituants à conserver pour le résumé final. Le catégoriseur est entraîné sur un corpus de phrases et un ensemble d'attributs extraits des phrases. Ces attributs sont essentiellement de surface : genre de l'article, nombre de phrases dans l'article, position des phrases, présence des conjonctions de coordination, des démonstratifs, fréquence des termes, etc. Les constituants extraits sont ensuite rassemblés dans leur ordre original.

Les deux approches suivantes descendent au niveau des mots pour composer le résumé final.

[Oka & Ueda, 2001] créent un graphe acyclique orienté à partir du texte source, les sommets sont des mots ou des séquences de mots et les arrêtes des relations entre les mots. Les relations se voient attribuer un score (basé sur le produit $tf \times idf$ des mots des sommets de l'arc de cette relation). Un sous-graphe est ensuite extrait, il représente la relation principale du texte. Quelques relations sont incluses dans le graphe afin d'ajouter des détails. Les mots présents dans le sous-graphe résultant sont ensuite mis bout à bout, dans le même ordre que dans le texte source, pour former une phrase résumé.

[Wan *et al.*, 2003] se concentrent sur la production de phrases titre. Les auteurs se soucient du contexte dans lequel les mots extraits se trouvent afin de ne pas rassembler des mots hors-contexte. La technique utilisée se base sur la décomposition en valeurs singulières pour tenir compte de la distribution des mots et des phrases afin de regrouper les phrases touchant au même thème. Un apprentissage automatique est préalablement effectué sur un corpus de documents, en se basant sur la correspondance entre les mots utilisés dans le titre du document et ceux utilisés dans le corps du document.

Ces trois techniques ne produisent que de courts résumés dont la cohérence grammaticale est mise en cause car la concaténation de constituants ou de mots pris dans le texte a peu de chances d'être grammaticale.

2.3.1.2 Marqueurs lexicaux de segments textuels

Facilement exploitables une fois la bonne ressource lexicale constituée, les marqueurs lexicaux peuvent être utilisés pour déterminer l'importance d'un segment textuel, soit directement (par proximité co-textuelle), soit indirectement, si le marqueur détecté aide à identifier une structure du texte. Dans ce dernier cas, les marqueurs contribuent dans une analyse au niveau structurel, mais ne peuvent à eux seuls déterminer intégralement la structure. Ils restent une information de surface car directement localisables.

Voici plusieurs utilisations classiques des marqueurs lexicaux :

- localisation des segments textuels généralement importants (introduction, résumé, conclusion...) ou peu importants (exemple, reformulation, explication...) ;
- détermination de la chronologie des événements, [Mani & Wilson, 2000] utilise des marqueurs comme *since* ou *until* ;
- délimitation des thèmes, [Farzindar & Lapalme, 2005] utilise des marqueurs linguistiques comme *analysis*, *decision* ou *discussion* dans des textes juridiques) ;
- identification des propos saillants, les auteurs de [Minel *et al.*, 2001] utilisent des ressources linguistiques constituées de « marqueurs discursifs explicites (morphèmes, mots, expressions et locutions...) caractéristiques d'une intention pragmatique de l'auteur du texte », et considèrent que « le jugement d'importance est fondé essentiellement sur ce que l'auteur a lui-même explicitement mis en valeur dans son texte ».

L'exploitation seule des marqueurs lexicaux ne suffit pas pour produire un résumé. En effet ils ne marquent que quelques traits sémantiques des textes, de manière ni systématique, ni unique.

Par exemple, la détection des exemples dans un document en français peut être aidée par des marqueurs lexicaux comme *par exemple*, *ainsi* ou *comme*. Cependant de tels marqueurs peuvent se situer au début, à la fin ou au beau milieu de l'exemple, voire même être absents, selon la construction du texte. L'information fournie par ces marqueurs est donc partielle.

2.3.1.3 Position de segments textuels dans le texte

Selon le genre du texte, il est souvent facile de connaître la structure globale du texte à savoir la position des titre, introduction, résumé, conclusion... Ces parties du document peuvent se révéler très utiles au résumé de texte par leur contenu informationnel naturellement important. Une fois localisées, elles peuvent être conservées pour le document cible, ou utilisées comme référentiel d'importance pour estimer celles d'autres parties.

D'autres positions peuvent être exploitées telles celles d'éléments d'une structure du texte au sein de cette même structure (par exemple la position d'un constituant d'une

phrase dans l'arbre syntagmatique de cette phrase), cependant ce travail s'effectue au niveau de l'analyse structurelle et non pas de surface.

2.3.1.4 Nature des constituants

C'est une information élémentaire et peu utilisée comme principale donnée, cependant certaines rares approches se basent essentiellement dessus pour produire des résumés.

Par exemple [Grefenstette, 1998] utilise la nature des syntagmes et propositions pour estimer leur importance, puis supprime les moins importants (après avoir établi un ordre d'importance en fonction de la nature) pour produire les phrases compressées. La cohérence obtenue est évidemment faible mais suffisante pour l'application souhaitée qui est la réduction de textes télégraphiques destinés à être lus pour les malvoyants.

2.3.2 Analyse des entités nommées

Nous appelons *entités nommées* (de l'anglais *named entity*) l'ensemble des noms de personnes, d'entreprises et de lieux présents dans un texte donné. Sont souvent associés à ces éléments d'autres syntagmes comme les dates, les unités monétaires ou les pourcentages, qui sont repérables par techniques similaires à celles utilisées pour la reconnaissance des entités nommées.

Les approches utilisant l'analyse des entités tentent de construire une représentation interne des différentes entités présentes dans le document et de dresser leurs relations. Ces relations sont ensuite utilisées pour repérer des motifs de connectivité, eux-mêmes utilisés dans la détermination de l'importance des segments textuels impliqués. Leur emploi est donc proche de celui des informations de surface, on retrouve naturellement les mêmes problèmes de cohérence structurelle lorsqu'elles sont utilisées comme seule information d'analyse.

De telles relations utiles au résumé automatique peuvent être la similarité lexicale (section 2.3.2.1), la similarité thématique (section 2.3.2.2) ou la coréférence (section 2.3.2.3).

2.3.2.1 La similarité lexicale

Elle permet de mesurer le niveau de similarité sémantique entre deux acceptions d'items lexicaux. Ainsi, plus des acceptions partagent des caractéristiques sémantiques communes, plus leur similarité est élevée.

Le principe de similarité lexicale est fréquemment utilisé dans les approches de résumé automatique [Barzilay & Elhadad, 1997, Boguraev & Neff, 2000, Chaves, 2001] [Fuentes & Rodríguez, 2002, Alonso & Fort, 2003] pour localiser des chaînes lexicales, c'est-à-dire des ensembles d'entités présentes dans le document et révélant une simila-

rité lexicale élevée. Une fois ces chaînes extraites, les plus importantes sont déterminées (par exemple celles qui ont des relations avec les mots du titre) et les segments textuels contenant le plus d'entités appartenant à ces dernières sont considérés comme les plus importants.

2.3.2.2 La similarité thématique

La proximité thématique exerce un lien entre deux entités comparable à celui que peut exercer la proximité lexicale. Des techniques similaires peuvent donc être réalisées pour utiliser cette information dans la détermination de l'importance des segments textuels. Il existe différentes approches en Traitement Automatique des Langues Naturelles (TALN) ou en recherche d'information pour extraire le thème d'un segment textuel comme les centroïdes [Radev *et al.*, 2004], la signature de sujets [Lin & Hovy, 2000], l'indexage sémantique latent [Deerwester *et al.*, 1990], les vecteurs sémantiques [Chauché *et al.*, 2003] ou les vecteurs conceptuels [Schwab *et al.*, 2005].

2.3.2.3 La coréférence

La coréférence est la référence dans une expression au même référent dans une autre expression. Il est bien question d'entités ici aussi, et il est aussi possible de dresser des chaînes appelées chaînes de coréférence utilisées de manière similaire aux chaînes lexicales. Des approches comme [Baldwin & Morton, 1998, Azzam *et al.*, 1999, Harabagiu *et al.*, 2003] utilisent les chaînes de coréférence pour le résumé automatique.

Un type de coréférence particulier est l'anaphore. Une anaphore est un « procédé consistant à rappeler un mot ou un groupe de mots précédemment énoncé par un terme grammatical » [TLF2007]. Ce groupe de mots désigne une entité. Si cette entité vient à être supprimée du document, ses référents (les termes grammaticaux) perdent leur sens et une incohérence sémantique profonde est engendrée. La résolution des anaphores a pour but de lier les entités à leur(s) référent(s), afin de mettre en place un système visant à éviter ce genre de problème. On pourra par exemple empêcher la suppression de l'entité référée, ou alors la supprimer avec ses référents, ou encore la supprimer et remplacer ses référents par l'entité même. Nous utilisons une gestion d'anaphores dans notre approche, décrite dans le chapitre 4.

2.3.3 Analyse de la structure du texte et de la phrase

L'analyse de la structure peut être utilisée pour déterminer l'importance d'un élément structurel en fonction de sa place et son rôle dans la structure même. À partir de cette information, les résumeurs peuvent réaliser des résumés par compression structurelle du

document. La production de tels résumés s'oppose à celle par extraction indépendante de segments textuels, laquelle ne tient pas compte des propriétés de cohérence structurelle du texte pour générer le résumé, elle se base généralement uniquement sur des éléments d'information non structurels[14].

Les segments textuels dont la structure est analysée s'échelonnent en taille du constituant de la phrase au texte complet. Il existe de nombreuses informations qui peuvent être utilisées pour déterminer une structure dans un document, nous en présentons ici quelques-unes des plus exploitées dans le domaine : le thème (section 2.3.3.1), l'événement (section 2.3.3.2), la relation rhétorique (section 2.3.3.3), le rôle thématique (section 2.3.3.4) et la syntaxe des phrases (section 2.3.3.5). Cette dernière est la seule à se situer exclusivement à un niveau de granularité de dépassant jamais la phrase, elle est donc généralement exploitée pour la production de résumés intra-phrastiques. Faisant l'objet de notre travail, c'est naturellement celle que nous approfondirons le plus. Quant aux autres informations, elles sont exploitées à un niveau de granularité généralement plus grand que la phrase, orientant alors habituellement la production vers un résumé inter-phrastique.

Dans le cas où le récepteur souhaiterait orienter la production du résumé selon certains critères (cf. section 2.2.2), il peut être particulièrement judicieux d'utiliser une structure correspondante, si elle existe. Par exemple si le récepteur souhaite résumer un document selon un certain thème, l'analyse de la structure thématique sera très adaptée pour produire le résumé. Le choix du type d'analyse structurelle se révèle donc très important selon le type d'application de résumé à réaliser.

Pourquoi la structure ? Une structure du document peut être exploitée pour différents objectifs. Généralement elle aide à identifier des segments textuels (les éléments structurels) importants ou peu importants afin de, respectivement, les conserver ou les éliminer. La décision est réalisée grâce à des propriétés connues de la structure quant à l'importance de ces différents éléments.

Par exemple, pour la structure rhétorique d'un texte, dans le cas d'une relation d'exemplification identifiée, en s'appuyant sur [Mann & Thompson, 1987] on dispose de l'information que le segment correspondant à l'exemplifiant est moins important que celui correspondant à l'exemplifié, ce qui peut permettre d'ôter le premier sans grande perte d'information.

Un autre objectif peut être de chercher à conserver une cohérence structurelle dans le résumé. En effet, une fois la structure identifiée, le résumeur peut utiliser des propriétés connues de cette structure pour ôter les segments textuels qui, en plus d'être peu importants sémantiquement, ne sont pas indispensables à la cohérence de cette structure. Cependant, il se peut que, tout en conservant une cohérence vis-à-vis de cette structure,

[14] informations de surface ou informations sur les entités

une incohérence vis-à-vis d'une autre soit engendrée suite à la suppression d'un segment textuel.

Par exemple, la suppression d'un élément thématique peut causer une incohérence de type rhétorique (voir 2.3.3.3). Prenons un texte qui traite d'un thème particulier, mais dont l'introduction introduit le sujet par une métaphore sur un thème différent. Si le résumeur décide de conserver uniquement le thème de la métaphore, alors une incohérence rhétorique (et non thématique) pourra être engendrée dans le cas où l'introduction serait en relation d'arrière plan avec le corps du texte et qu'elle est le satellite de cette relation (donc la partie à supprimer).

2.3.3.1 Structure thématique

S'il est possible de déterminer une telle structure, alors il devient possible de conserver uniquement les segments textuels appartenant au(x) thème(s) choisi(s) par le récepteur ou jugé(s) le(s) plus important(s) de manière générale. Comme nous l'avons vu en section 2.3.2.2, il est possible de réaliser cette tâche par la recherche de liens entre entités. D'autres approches utilisent des espaces vectoriels pour représenter le thème des segments textuels S (des mots, phrases, textes...). Les dimensions de ces vecteurs sont des mots ou alors des concepts activés par S. L'orientation du vecteur dans l'espace peut alors être interprétée comme un thème. Enfin des techniques géométriques peuvent être utilisées pour comparer les vecteurs, et donc les thèmes.

Par exemple, [Ando *et al.*, 2000] utilisent une technique proche du procédé de décomposition en valeurs singulières (*Singular Value Decomposition, SVD*), elle-même utilisée dans l'indexation sémantique latente (*Latent Semantic Indexing, LSI*) de [Deerwester *et al.*, 1990], ou encore [Hirao *et al.*, 2002] se fonde sur les *Support Vector Machines* pour séparer les phrases clés des autres.

2.3.3.2 Structure événementielle

La majorité des approches traitant des événements se concentre sur un seul événement par document et tente de remplir un patron 2.2.6.1 pour ensuite générer un court résumé. Cependant l'information événementielle commence à être étudiée d'un point de vue structurel dans le but d'extraire des éléments structurels correspondant à des périodes temporelles et/ou spatiales jugées plus importantes.

Par exemple [Mani, 2004] aborde la problématique du résumé de textes narratifs, en s'appuyant principalement sur des indices temporels et événementiels. Il étudie les évènements sur trois plans : la scène, l'histoire et l'intrigue, dans le but d'extraire les évènements clés, scènes clés, et les intrigues saillantes. Il compte sur les méthodes actuelles (basées sur le marquage lexical, l'étude de la structure rhétorique, l'analyse morpho-syntaxique...)

et futures pour extraire de tels indices temporels.

2.3.3.3 Structure rhétorique

Elle est déterminée par les relations rhétoriques présentes au sein du document.

Ces relations sont étudiées par les auteurs de [Mann & Thompson, 1987] qui proposent une théorie, la Rhetorical Structure Theory (RST), dans laquelle ils élaborent une typologie précise des relations, basée sur un modèle en termes de noyau et de satellite, et sur la spécification des interactions que ces relations établissent entre les éléments en présence. Leur étude ouvre la voie à une étude computationnelle des relations rhétoriques.

Cette « RST propose une explication de la cohérence des textes[15]. [...] Le but de la RST est de décrire les textes, plutôt que les processus qui sous-tendent leur création et leur interprétation. Elle postule un ensemble de possibilités de structures — divers types de "blocs de construction" — dont on peut observer les occurrences dans les textes. Ces "blocs" se situent à deux niveaux, le principal ayant trait à la "nucléarité" et aux "relations" (souvent appelées relations de cohérence dans la littérature linguistique). »[16] Nous ne présentons pas ici le second niveau de structures, les *schémas*, car ils ne sont pas utilisés en résumé automatique.

Le noyau d'une relation étant par définition plus important que le satellite, une technique de résumé peut consister à conserver uniquement les noyaux dans le document cible. La principale difficulté est de déterminer correctement la structure rhétorique (SR).

La principale approche visant cet objectif est celle de [Marcu, 1998], dans laquelle l'auteur utilise une combinaison d'heuristiques standard pour aider au choix de la bonne SR du texte source, au niveau inter-phrase et intra-phrase. Les sept métriques suivantes sont utilisées :

- groupement par thème : pour deux nœuds frères de l'arbre de la SR, leurs feuilles doivent correspondre au mieux avec les frontières de changement de thèmes ;
- utilisation des marqueurs : si des marqueurs sont présents dans le texte source, la SR doit les vérifier au mieux ;
- groupement rhétorique par thème : identique à la première métrique si ce n'est que la comparaison se fait avec les noyaux des relations et non les feuilles ;
- poids des branches situées à droite : sont préférés les arbres dont les branches droites sont plus importantes, car ce sont habituellement ces branches qui contiennent les ajouts de l'auteur (moins importants et donc supprimables) ;

[15] la cohérence consiste en l'absence d'illogismes et de lacunes. C'est-à-dire que pour toute partie d'un texte cohérent, il existe une fonction, une raison plausible à sa présence, qui soit évidente pour les lecteurs, et par ailleurs, le lecteur n'a pas le sentiment que des parties manquent à l'ensemble

[16] `http://www.sfu.ca/rst`

- similarité avec le titre : sont préférés les arbres dont les unités saillantes (noyaux) sont les plus similaires au titre du texte ;
- position des phrases : les phrases en début ou fin de paragraphe/document sont habituellement considérées comme plus importantes ; une mesure de similarité, du même type que pour la métrique précédente, est alors effectuée ;
- connexion des entités : l'information sur les relations entre les mots est prise en compte, par exemple avec les chaînes lexicales.

Selon le poids de chaque métrique utilisée dans l'heuristique, le traitement est plus efficace pour différents genres de documents, ce qui tend à renforcer l'idée qu'un corpus ayant un genre donné, comme unité d'évaluation n'est pas discriminant. L'auteur n'est pas parvenu à trouver une solution fonctionnant pour tout genre de texte. Il aurait pu se préoccuper de rechercher des intervalles de valeurs pour calibrer son système, mais il n'a malheureusement pas poussé la discussion jusque là.

Une fois la SR déterminée, un ordre partiel entre les différents satellites est établi, les satellites plus proches de la racine se voient attribuer une importance plus grande. Les satellites sont ensuite supprimés, des moins importants aux plus importants selon la taille du résumé désirée. La cohérence est assez bien conservée dans les cas où l'analyse de la SR est correcte, cependant cet objectif n'est que très partiellement atteint.

Une relation rhétorique peut aussi bien lier des constituants, que des phrases ou des paragraphes. Les deux dernières relations que nous allons maintenant aborder se situent exclusivement à l'échelle des constituants.

2.3.3.4 Structure des rôles thématiques des phrases

Nous appelons rôle thématique une relation sémantique entre un prédicat (ex : un verbe) et un argument (ex : un groupe nominal) d'une phrase. L'utilisation des rôles thématiques est très peu commune dans le résumé automatique. Les rôles thématiques incluent l'agent, l'*experiencer*, le patient, l'instrument, la cause, le lieu, le but et la source.

Selon le rôle thématique auquel appartient un constituant, son importance grammaticale ou informationnelle peut varier.

[Jing, 2000] utilise cette information parmi d'autres pour compresser des phrases. Les ressources qu'il exploite sont : un corpus contenant des phrases et leur forme compressée correspondante, écrite par des humains ; un lexique incluant des sous-catégorisations de verbes, utilisé pour déterminer les arguments indispensables des verbes des propositions ; la base de données lexicale Wordnet [Miller *et al.*, 1990] contenant des relations lexicales (synonymie, antonymie, méronymie...) entre les mots ; le parseur *English Slot Grammar* (ESG) [McCord, 1990] qui annote les constituants des phrases avec leur nature et leur rôle thématique. Les différentes étapes de l'algorithme sont : le parseur ESG

produit l'arbre syntagmatique des phrases ; les constituants indispensables à la cohérence grammaticale sont déterminés à l'aide des informations sur leur rôle thématique et sur les sous-catégorisations des verbes les précédant dans la phrase ; le système décide quels sont les constituants les plus proches du thème du document à l'aide de Wordnet : les relations lexicales entre les mots sont extraites, plus la connexité d'un mot avec les autres de son contexte local est élevée, plus il est considéré comme proche du thème du contexte local ; une probabilité d'être supprimé est attribuée à chaque constituant des phrases, en utilisant le modèle de Bayes, à partir du corpus de phrases et leur version compressée, en fonction des verbes utilisés et du rôle thématique des constituants ; la réduction des phrases est ensuite effectuée en fonction des annotations précédentes, après une pondération des différents facteurs d'importance et de cohérence, et un seuil fixé en fonction de la qualité du résumé désirée. L'évaluation de leur système, faite par les auteurs, a montré que les choix de suppression des constituants concordaient à peu près à 81 % avec ceux faits par des humains. La taille des textes est réduite d'environ 33 % par le système, contre 42 % par des humains. Cette approche a l'avantage de pondérer l'importance des constituants par des données contextuelles. La perte majeure d'information est fortement dépendante du modèle d'apprentissage probabiliste utilisé, qui exploite le rôle thématique des constituants pour déterminer leur importance.

Le rôle thématique se limitant aux constituants attachés au prédicat, seuls ces derniers font l'objet d'une suppression. Les constituants attachés aux noms, adjectifs et prépositions ne le font pas, réduisant ainsi les possibilités de compression.

2.3.3.5 Structure syntaxique des phrases

Enfin, une des dernières structures utilisée dans le domaine est la structure syntaxique. Son étude tient compte, notamment, de la catégorie[17] et de la fonction syntaxique des constituants des phrases.

Extraire la catégorie lexicale d'un mot, ainsi que certains traits basiques de certaines catégories tels le genre et le nombre pour les noms et adjectifs, est une tâche réalisée avec une bonne fiabilité de nos jours, il en va de même à l'échelle du constituant. Cependant, déterminer les arbres syntagmatiques des phrases se révèle bien plus difficile. Les systèmes visant à effectuer cette tâche n'y parviennent que très partiellement, ne restituant que très approximativement l'imbrication et l'attachement des constituants entre eux.

Les approches de résumé automatique exploitant des données syntaxiques ne considèrent que peu d'information parmi l'ensemble des traits morphologiques et syntaxiques. Les deux principales informations exploitées sont la catégorie des constituants et leur po-

[17]Par catégorie des constituants, nous entendons la catégorie lexicale de leur tête (voir prochain chapitre, section 3.2.1).

sition dans l'arbre syntagmatique, comme nous allons le voir à travers les deux prochaines approches présentées.

[Knight & Marcu, 2002] abordent le problème la compression de phrases sous deux angles : un modèle probabiliste et un modèle basé sur la décision. Les deux modèles utilisent le parseur Collins, [Collins, 1997], qui fournit des informations sur la nature des syntagmes de phrases sous forme d'arbres syntagmatiques.

Le premier emploie un modèle de canal bruité (*noisy-channel model*) qui consiste à faire l'hypothèse : « la phrase à comprimer fut autrefois courte et l'auteur y a ajouté des informations supplémentaires (le bruit). » Le but est alors de retrouver ces informations pour les supprimer. Le modèle probabiliste est bayésien, et les auteurs l'entraînent sur un corpus de documents avec leur résumé produit par un humain. Le moteur d'apprentissage a pour but de sélectionner les mots à conserver dans la phrase comprimée. Une faible probabilité est attribuée à une phrase comprimée lorsque cette dernière est incorrecte grammaticalement ou a perdu certaines informations comme la négation. Pour réaliser leur évaluation, les auteurs ont créé un corpus de test en extrayant 32 paires de phrases (phrase originale, phrase résumée) de leur corpus. Les autres paires de phrases (au nombre de 1035) constituaient le corpus d'entraînement. Leur métrique est fondée sur un score de bi-grammes de caractères. La justesse grammaticale obtenue n'est pas suffisamment bien conservée dans la plupart des cas et une légère perte d'*information importante* est à noter.

Le second modèle utilise des règles de transformation appliquées aux arbres syntagmatiques des phrases du texte dans le but de réduire ces arbres puis de recomposer des phrases plus courtes. Les règles sont composées d'un ensemble d'opérations élémentaires de manipulation des arbres. Le moteur d'apprentissage chargé de créer ces règles utilise le programme C4.5 [Quinlan, 1993]. Le taux de compression n'est pas paramétrable et avoisine les 55 % dans l'expérimentation. La cohérence grammaticale et la conservation de contenu important (l'absence de perte majeure d'information) sont légèrement inférieures au premier modèle des auteurs.

Les auteurs de [Hovy *et al.*, 2005] utilisent principalement deux informations syntaxiques pour compresser des phrases. Un arbre syntagmatique et des relations de dépendances qu'ils appellent *éléments basiques* (*Basic Elements* ou BE en anglais). Les auteurs définissent les BE comme étant soit « la tête d'un constituant syntaxique majeur (nom, verbe, adjectif ou expression adverbiale) », soit « une relation entre un BE-tête et un simple dépendant, exprimée par un triplet (tête, modifieur, relation). » Ces BE sont extraits automatiquement d'arbres syntagmatiques des phrases[18], grâce à un outil développé

[18]Ces arbres sont produits par l'analyse Collins [Collins, 1997].

par les auteurs de l'approche ainsi que par J. Fukumoto, mais dont aucun article sur la technique utilisée n'est accessible[19].

Pour compresser les phrases, les auteurs commencent par extraire automatiquement les BE des phrases, puis à classer ces triplets par importance, en fonction d'un score basé sur un rapport de vraisemblance [Dunning, 1994]. Ensuite des arbres de dépendances sont dressés à partir de l'ensemble des BE extraits, et différents élagages sont testés. Pour chaque phrase, est conservé l'arbre élagué qui maximise les trois critères suivants :

- l'arbre élagué est un des plus petits ;
- l'arbre élagué a obtenu une des meilleures probabilités de génération selon les règles d'une Grammaire Probabiliste Hors Contexte (GPHC) basée sur le corpus étiqueté *Penn TreeBank* ;
- l'arbre élagué ne contient que des BE bien classés (importants).

Les résultats de leur validation montrent un gain d'environ 8 % de rappel, lorsque leur technique de compression de phrases est appliquée sur des résumés provenant de la tâche (à la conférence DUC2003) qui consiste à produire un résumé de 100 mots par extraction.

Le principal critère de compression de cette approche repose sur le modèle probabiliste de la GPHC, ce qui rend le système de compression très dépendant de ce modèle ainsi que du corpus qui permet de générer les probabilités des règles de ce modèle. Cette remarque s'applique aussi à la précédente approche, pour le modèle bayésien.

2.4 Les modèles de résumé automatique des logiciels commerciaux

Avant de terminer ce chapitre, nous présentons maintenant un rapide aperçu de quelques solutions commerciales de résumé automatique. Comme nous avons pu le voir dans les précédentes sections, la qualité des résumés automatiques produits à l'heure actuelle ne peut satisfaire à tout type d'application. Par exemple, obtenir des résumés par reformulation est quasiment innaccessible aujourd'hui, ou encore conserver la cohérence des différentes structures du texte n'est pas toujours réalisable.

Les logiciels commerciaux doivent proposer des solutions robustes afin de justifier leur prix d'achat, ils ne peuvent alors s'autoriser à s'aventurer dans des techniques peu fiables. Ainsi, tous les résumeurs automatiques commerciaux pour lesquels nous avons pu obtenir des informations sur leur fonctionnement se cantonnent au résumé par extraction de phrases, évitant alors toute reformulation et toute modification de phrase. Ces logiciels se

[19] Les auteurs citent une référence, datée de 2005, qui est une adresse Internet invalide.

démarquent toutefois entre eux par l'information exploitée pour déterminer l'importance des phrases.

Le tableau 2.1 présente 5 solutions commerciales qui se démarquent sur Internet. Les descriptions sont souvent imprécises car très peu d'informations sont révélées par les entreprises qui développent ces logiciels, certainement pour des raisons de propriété intellectuelle.

Nom	**Information exploitée**	**Autres caractéristiques**
Pertinence[a]	marqueurs lexicaux discursifs	technique fortement dépendante du genre, adaptée aux documents scientifiques ou techniques, mais pas aux narratifs
Copernic[b]	inconnue	moteur d'apprentissage basé sur un modèle bayésien pour déterminer les mots clés du sujet
Intellexer[c]	nature des relations verbe–objet (appelées concepts)	l'utilisateur peut intervenir sur le choix des concepts importants
TextAnalyst[d]	concepts issus d'un réseau sémantique produit à partir du document à résumer	
Extractor[e]	inconnue[f]	

Tab. 2.1 – 5 solutions commerciales de résumé automatique.

[a]Site Internet de Pertinence : `http://www.pertinence.net`
[b]Site Internet de Copernic : `http://www.copernic.com/fr/products/summarizer/`
[c]Site Internet de Intellexer : `http://summarizer.intellexer.com/`
[d]Site Internet de TextAnalyst : `http://www.megaputer.com/textanalyst.php`
[e]Site Internet de Extractor : `http://www.extractor.com/`
[f]Le descriptif technique se limite à : « *Extractor is an exceptional content summarization utility using patented technology to summarize text, e-mail and HTML content into weighted lists of keywords and keyphrases.* » (Extractor est un utilitaire exceptionnel de résumé de contenu utilisant une technologie brevetée pour résumer des textes, e-mails et pages HTML en des listes pondérées de mots clés ou phrases clés.)

2.5 Conclusion

Dans ce chapitre nous avons fait un tour d'horizon de la majorité des types d'approches de résumé automatique. Dans une première partie nous avons présenté les principales caractéristiques qui font la diversité des résumés, du document source au document cible, de la granularité des segments textuels analysés au type de processus de production. Une fois ces caractéristiques prises en compte, les approches doivent choisir quels types d'information seront traités dans l'analyse. C'est ce que nous avons décrit en section 2.3. Nous avons choisi de découper ces types d'informations en trois types : informations de surface, informations au niveau entité et informations structurelles.

Notre attention s'est particulièrement portée sur les approches exploitant des informations structurelles et particulièrement celles qui produisent des résumés par compression structurelle car ces derniers ont l'avantage sur les résumés par extraction indépendante de segments textuels de conserver une cohérence structurelle, propriété importante pour notre approche. Ces derniers ont été et sont encore largement étudiés dans la littérature, mais possèdent cette limitation intrinsèque à leur processus de production, nous ne nous y intéressons pas davantage dans ce travail. Quant à la qualité de ceux produits par compression structurelle, elle dépend grandement du type de structure choisie.

Exploiter les structures dont les unités sont de taille moyenne supérieure à la phrase (thématiques, événementielles, rhétoriques. . .) pose des difficultés de cohérence du résumé produit, car même lorsque les éléments structurels sont bien identifiés, les mettre bout-à-bout pour produire le résumé aboutit que rarement à un texte cohérent. Les différents éléments extraits doivent être correctement liés les uns aux autres afin de conserver une continuité discursive, tâche difficilement abordable à l'heure actuelle. De plus, la granularité moyenne ou grande de ces structures les rend difficilement identifiables avec précision.

La structure syntaxique des phrases, de granularité plus petite que celle des précédentes structures, est davantage abordable de nos jours. Les analyseurs syntaxiques récents parviennent à des résultats qui peuvent s'avérer exploitables pour un résumé automatique par compression de phrases, comme nous le verrons au chapitre 4.

Les rares approches comme [Knight & Marcu, 2002, Hovy *et al.*, 2005] qui utilisent une telle information dans ce but se basent sur des modèles probabilistes pour déterminer si la phrase compressée est un bon candidat. Ils tentent de retrouver automatiquement des propriétés linguistiques d'importance et de cohérence à partir d'informations statistiques de corpus. Les limitations de telles approches sont doubles. D'une part les apprentissages sont approximatifs, cela étant dû aux modèles naturellement imparfaits et aux corpus insuffisamment couvrants. En effet, si l'apprentissage porte sur un corpus de texte associé à des résumés produits par des humains ([Knight & Marcu, 2002]), alors, le corpus étant inévitablement orienté vers une certaine thématique, un certain genre, une certaine spécialisation du vocabulaire. . . , la méthode de compression sera circonscrite aux cas du corpus d'apprentissage. Si l'apprentissage détermine des constructions syntaxiques plausibles [Knight & Marcu, 2002, Hovy *et al.*, 2005], alors les transformations syntaxiques effectuées seront circonscrites à celles rencontrées dans le corpus d'apprentissage. D'autre part les informations utilisées se limitent généralement à la position des constituants dans la phrase alors que d'autres propriétés linguistiques peuvent se révéler déterminantes dans l'estimation de l'importance de ces constituants, notamment les traits de sous-catégorisation des lemmes.

Ces constatations dévoilent des pistes inexplorées dans la compression de phrases que nous décrivons dans le prochain chapitre.

3

La compression de phrases par élagage de l'arbre syntagmatique

Sommaire

3.1 Introduction 37
3.2 Une classification des éléments effaçables 43
3.3 Exploitation de traits linguistiques dans notre compression de phrases 61
3.4 Esquisse d'un modèle computationnel 77
3.5 L'influence du genre de texte sur l'importance des modifieurs et compléments 83
3.6 Les limites de la localisation du contenu important 84
3.7 Conclusion 85

3.1 Introduction

Le point de départ de notre approche fut l'intuition que, pour la compréhension d'un texte, **la fonction syntaxique des constituants des phrases est un facteur conséquent dans l'évaluation de l'importance de ces constituants**. Nous prenons cette intuition comme hypothèse principale à notre approche. Elle prend ses racines dans l'analyse grammaticale logique classique et dont on trouve des manuels connus comme [Mauffrey & Cohen, 1995, Bes1990, Wagner & Pinchon, 1962], [Grevisse, 1993–1997].

Afin de mieux appréhender notre hypothèse, nous avons réalisé une étude sur un petit corpus de textes, en s'intéressant aux informations syntaxiques impliquées dans l'importance des constituants des phrases. L'étude a consisté en une analyse manuelle et minutieuse de l'importance des constituants selon leurs propriétés syntaxiques, à travers

en ensemble de documents . Les textes furent pris dans différents genres textuels. Par exemple, des articles journalistiques du quotidien « Le Monde », des articles en biologie et bio-informatique et un conte polynésien (voir l'annexe B.1). Le but de cette démarche était d'identifier un maximum de cas syntaxiques autorisant une suppression de constituant sans dégradation majeure de la compréhension du texte, pour ensuite extraire et factoriser de ces bons candidats les caractéristiques syntaxiques impliquées dans la propriété d'effacement. La fonction fut l'information syntaxique qui s'est révélée être la plus fiable pour déterminer l'importance des constituants.

Ainsi, ne sont pas toujours indispensables pour comprendre le sens principal de la phrase, certains épithètes, certains compléments circonstanciels, *etc.*

Par exemple, dans la phrase :

Exemple 3.1 *Un chat gros et laid mange une souris.*

le groupe adjectival épithète *gros et laid* peut être supprimé sans nuire réellement à la compréhension, et à l'intérêt (la phrase ainsi contractée ne nuit pas *a priori* à la cohésion sémantique du texte réduit).

La théorie du gouvernement et du liage de Noam Chomsky, *Government and Binding* [Chomsky, 1981, Chomsky, 1982, Chomsky, 1986], désormais GB, confirme notre intuition initiale. Étroitement liée à la fonction syntaxique, elle décrit des relations de gouvernance syntaxique entre certains mots de la phrase, les têtes (gouverneurs), et certains groupes de mots, les constituants gouvernés (voir section 3.2.1). Les têtes sont indispensables aux cohérences syntaxique et sémantique des constituants dans lesquelles elles s'inscrivent alors que les constituants qu'elles gouvernent peuvent, selon les cas, être effacés sans nuire à la cohérence syntaxique, et parfois même sans grande perte sémantique. Ainsi, dans l'exemple 3.1, le groupe adjectival épithète *gros et laid* est un constituant gouverné par la tête *chat*.

Alors que nos premières approches exploitant GB [Yousfi-Monod & Prince, 2005b, Yousfi-Monod & Prince, 2005a, Yousfi-Monod & Prince, 2006] s'appuyaient uniquement sur l'influence en importances syntaxique et sémantique du gouvernement, nous proposons ici une approche bien plus détaillée et enrichie, dans laquelle nous explorons la structuration précise des phrases, nous décrivons la connexion des constituants selon leur fonction syntaxique et nous étudions leur importance selon un ensemble de traits linguistiques.

Ainsi ce chapitre qui présente notre étude linguistique s'appuie sur une représentation chomskienne de la structure syntaxique des phrases : des arbres syntagmatiques et des relations de gouvernement entre mots et constituants. Un autre modèle syntaxique exploité en TALN est celui des grammaires de dépendance de Lucien Tesnière [Tesnière, 1934, Lucien, 1959]. Le principe fondamental de ces grammaires est que chaque

mot dépend d'un autre dans la phrase. Cette notion est étroitement liée à celle de gouvernance, où les têtes syntaxiques entretiennent des relations du même ordre. Les deux modèles peuvent constituer le support à une approche de compression syntaxique comme nous l'avons vu au chapitre précédent :

- [Knight & Marcu, 2002] pour la grammaire de constituants ;
- [Hovy *et al.*, 2005] pour la grammaire de dépendance.

Nous avons cependant fait le choix d'exploiter le premier modèle car l'analyseur syntaxique utilisé pour notre mise en œuvre et que nous présentons dans le prochain chapitre s'appuie fortement sur la grammaire chomskienne tout en nous offrant l'avantage d'une grande accessibilité, car maintenu par un membre de notre équipe au sein de notre laboratoire. Cet avantage nous a permis, à travers une interaction fréquente et efficace avec le concepteur, d'adapter le modèle syntaxique de cet analyseur à notre grammaire définie dans ce chapitre.

L'étude du vaste domaine du résumé automatique (chapitre précédent) nous a permis de préciser notre approche, en la définissant en deux objectifs précis : produire un résumé automatique **conservant la structure du texte**, et **indépendant d'un modèle d'apprentissage et d'un corpus particulier**. Le premier objectif nous a orienté vers la compression syntaxique : c'est un résumé par compression structurelle pour lequel la production de textes cohérents structurellement est abordable actuellement. Le second objectif nous oriente vers l'exploitation de théories structurelles (comme GB).

3.1.1 La compression de phrases

Dans ce chapitre, nous allons donc développer une approche théorique de compression de phrases. Compresser une phrase, dans le sens le plus général, signifie enlever certains mots de la phrase. De manière plus formelle, la compression de phrases peut se définir ainsi :

Définition 3.2 *Soient $P = \langle m_1, \ldots, m_n \rangle$ et $P' = \langle m'_1, \ldots, m'_k \rangle$ deux phrases, définies comme séquences de mots, avec m_i le $i$ème mot de la phrase P, m'_j le $j$ème mot de la phrase P'.*
P' est une compression de P si et seulement si $P' \subseteq P$.

Dans la théorie des langages, en considérant les mots comme les symboles de l'alphabet du langage, cela revient à dire que P' est un sous-mot de P, en considérant les mots de la phrase comme les symboles du langage.

Cette définition ne considère que le découpage en mots des phrases. Or, effacer des mots dans une phrase, sans se soucier des regroupements syntaxiques risque fort d'aboutir

à des phrases agrammaticales. C'est pourquoi la plupart des approches qui s'intéressent à la compression de phrases le font en se basant sur une unité syntaxique qui est le constituant.

Nous définissons les *constituants* à partir de la définition de syntagme de [TLF2007] comme un « groupe d'unités linguistiques significatives formant une unité dans une organisation hiérarchisée de la phrase ». L'organisation ici est la syntaxe de la phrase. Ainsi, peuvent être constituants un groupe nominal, un groupe verbal, un groupe prépositionnel, une proposition, *etc.* Par exemple, le constituant groupe nominal *un médecin de famille* est composé de deux constituants : un groupe nominal *un médecin* et un groupe nominal prépositionnel *de famille*.

Deux unités sont souvent considérées dans l'organisation de la phrase : les *syntagmes* et les *groupes*. Nous les considérons dans ce travail comme des synonymes de constituant.

3.1.2 La compression syntaxique de phrases

Compresser des phrases en supprimant des constituants revient à élaguer l'arbre syntagmatique des phrases, car les nœuds internes de ce dernier sont des constituants. D'autres types de compression syntaxique peuvent théoriquement exister, considérant d'autres opérations sur l'arbre syntagmatique, comme la fusion de branches par exemple. De telles approches peuvent être complémentaires à celles qui élaguent l'arbre syntagmatique, mais elles ne font pas l'objet de notre travail.

Il est maintenant question de compression syntaxique, car basée sur une structure syntaxique. La granularité du constituant, comme unité syntaxique, étant la plus répandue, nous généralisons notre définition de compression syntaxique au principe de suppression de constituants, vérifiant aussi la définition 3.2, et la définissons ainsi :

Définition 3.3 *Une phrase P' est une compression syntaxique d'une phrase P si l'arbre syntagmatique de P' est un arbre de même racine et un sous-graphe de l'arbre syntagmatique de P.*

Conserver la même racine permet de conserver le cœur de la phrase, c'est-à-dire le couple sujet – prédicat, lesquels se placent en fils directs de la racine et sont indispensables à la cohérence syntaxique.

Cette nouvelle définition n'autorise plus que la suppression de constituants, plutôt que de mots, augmentant alors les chances de produire des phrases syntaxiquement cohérentes. Cependant, la cohérence grammaticale n'est pas toujours garantie par cette contrainte de granularité dans la phrase. Par exemple si dans la phrase *Jean envoie une lettre à Marie* sont effacés les deux compléments du verbe, qui sont bien des constituants, alors la phrase devient agrammaticale.

Les approches actuelles en compression syntaxique de phrases ont pour base commune la définition 3.3. Les différences se situent alors sur les méthodes de choix des constituants à supprimer.

Enfin, à l'échelle du texte, compresser les phrases permet de produire un certain type de résumé : le résumé par compression de phrases, que nous définissons ainsi :

Définition 3.4 *Un texte* $T' = \{P'_1, P'_2, \ldots, P'_{n'}\}$ *est un résumé par compression de phrases d'un texte* $T = \{P_1, P_2, \ldots, P_n\}$ *ssi* $n' = n$ *et* $\forall i \in \{1..n\}$, $P'_i \subseteq P_i$, *avec* P'_i *et* P_i *des phrases situées en* i*ème position de leur texte respectif.*

Ainsi, nous appelons résumé de texte par compression de phrases un texte dont chaque phrase a été éventuellement compressée et jamais supprimée. Notre approche se concentrant sur la compression des phrases, nous utiliserons ce type de résumé de texte lors de l'évaluation de notre prototype de compresseur de phrases, définie au chapitre 5.

3.1.3 Nos objectifs

Nous venons de définir ce que nous appelons une compression syntaxique de phrases, en tenant compte, pour seule contrainte de qualité, de la conservation du constituant syntaxique. Nous décrivons maintenant nos deux objectifs en terme de qualité de compression produite.

Le but est de supprimer un maximum de constituants tout en conservant au mieux deux critères :

1. la cohérence structurelle des phrases, c'est-à-dire la cohérence grammaticale ;
2. la conservation du contenu informationnel important.

Une bonne compression syntaxique maximise ainsi la cohérence structurelle tout en minimisant la perte de contenu important. La structure syntaxique d'une phrase est cohérente si la phrase est grammaticale, c'est-à-dire si elle respecte les règles de grammaire de la langue dans laquelle elle est écrite.

Dans notre approche, la granularité maximale d'analyse est la phrase, car notre but est d'exploiter les informations syntaxiques des phrases pour réaliser notre compression. En restant à l'échelle de la phrase, nous ne considérons alors évidemment pas les informations contextuelles de la phrase. Ainsi notre étude du contenu informationnel important exclura délibérément de telles informations. De nombreuses autres approches se concentrent sur cet aspect, mais cette étude sort du cadre de notre approche. Dans notre cas, nous faisons l'hypothèse que des indices sur le contenu informationnel important sont présents au sein même de la phrase analysée. Ce sont ces informations que nous exploitons dans ce chapitre.

Durant tout ce chapitre, nous n'aborderons pas la problématique de la mise en œuvre de notre compresseur syntaxique en particulier, réservée au prochain chapitre. Nous resterons donc dans un cadre théorique général, et traiterons de problèmes conceptuels communs à tout système de compression automatique.

3.1.4 Notre compression syntaxique de phrase

Alors que certaines approches en compression de phrases, comme [Grefenstette, 1998], se concentrent principalement sur l'importance individuelle d'éléments de la phrase, en mettant en second plan la cohérence syntaxique, nous avons fait le choix de traiter le problème de la compression de phrases en visant en premier lieu la conservation de la cohérence syntaxique. Ainsi nous commençons, section 3.2, par déterminer une classification des constituants potentiellement effaçables sur le plan syntaxique[20], selon leur fonction. Nous utilisons le terme *effaçabilité* (et ses dérivés) pour désigner cette propriété d'effacement. Lorsqu'il s'agit de traiter de l'importance du contenu informationnel, nous parlons d'*importance*.

Dans cette section, nous étudions différentes grammaires existantes, à commencer par celle décrite dans GB, en orientant cette étude sur le caractère d'effaçabilité selon la fonction syntaxique. Cette étude aboutit à la définition de deux classes de fonctions de constituants, l'une, regroupant les éléments que nous appellerons *modifieurs*, qui sont systématiquement effaçables, et l'autre, regroupant les éléments que nous appellerons *compléments*, dont le caractère d'effaçabilité dépend de traits lexicaux de l'élément qu'ils complémentent.

À ce stade, nous disposerons déjà d'une méthode théorique de production de compressions syntaxiques, réalisable à deux niveaux de qualité différents :

- soit basée sur l'effacement des modifieurs, pour ainsi obtenir des phrases grammaticales ;
- soit basée sur l'effacement des modifieurs et des compléments, pour ainsi obtenir des phrases potentiellement agrammaticales vis-à-vis de leurs compléments, mais dont le taux de compression est plus élevé en moyenne.

Cela constitue un premier palier de compression.

Certains de ces modifieurs et compléments peuvent cependant être importants. Nous continuons alors notre approche, section 3.3, par une étude plus détaillée des modifieurs et compléments, basée sur l'exploitation d'autres traits linguistiques, et dont le but est d'améliorer cette compression, en fournissant des critères de détermination de l'importance des modifieurs et compléments, et ainsi limiter l'effacement de ceux qui sont importants. Nous aboutirons alors à un deuxième palier de compression, dont le taux de compression

[20] C'est-à-dire effaçables sans nuire à la cohérence syntaxique.

est inférieur au premier mais dont la conservation du contenu important est meilleure.

Plus nous avancerons dans l'analyse, plus nous proposerons de tels critères. L'estimation de l'importance de chaque constituant varie d'un critère à un autre. De plus, de nombreuses informations requises par ces critères sont difficilement (et souvent partiellement) extractibles de manière automatique à l'heure actuelle, nous incitant alors à proposer des heuristiques pour exploiter ces informations partielles et ainsi augmenter la robustesse du système de compression de phrases visé. Ainsi, l'importance estimée pour chaque constituant sera variable en intensité et fiabilité.

La compression d'une phrase peut alors être réalisée en fixant une limite de tolérance pour la valeur d'intensité et une autre pour la valeur de fiabilité, pour ensuite éliminer de la phrase les constituants qui respectent cette tolérance. Le procédé de compression (notre deuxième palier) devient alors paramétrable, selon la tâche souhaitée, proposant ainsi différents taux de compression associés à différents niveaux de conservation de l'information.

Certaines informations d'importance restent toutefois inaccessibles dans le cadre d'une analyse syntaxique de la phrase. Nous discutons de ces limites en section 3.6, et proposons une piste conceptuelle à explorer : une interaction entre un utilisateur et un système informatique de résumé automatique.

3.2 Une classification des éléments effaçables

Le test de suppression[21] des constituants est abordé par de nombreux ouvrages de grammaire française, dont ceux cités au début de ce chapitre, pour aider à la détermination de la fonction syntaxique d'un constituant. Le test est validé si la phrase résultante reste grammaticalement cohérente. Cependant, les textes linguistiques traitant de l'importance des constituants dans la phrase selon leur fonction syntaxique sont beaucoup plus rares. Des recommandations sont présentées par les linguistes ([Tomassone, 2001] pour les compléments circonstanciels par exemple), mais pas de règle fondamentale. Nous avons donc considéré ces recommandations comme des hypothèses de travail et nous avons cherché à les étayer empiriquement.

Notre étude a débuté en s'inspirant de la grammaire classique, notée GC, au sujet des fonctions syntaxiques des constituants, c'est-à-dire celle qui est traditionnellement enseignée dans les écoles françaises et qui décompose les phrases principalement en sujet, verbe et compléments. Les différentes fonctions syntaxiques qui nous intéressent dans cette grammaire ont été principalement les compléments circonstanciels, les différents

[21] Il est aussi appelé *effacement* ou *soustraction*. C'est une manipulation syntaxique qui consiste à supprimer un mot ou un groupe de mots afin de tester une propriété linguistique.

compléments de verbes, les épithètes, les compléments de nom, les appositions et les propositions relatives, car ces constituants semblaient être de bons éléments effaçables.

Puis, nous nous sommes rendu compte que certains constituants de différentes catégories que nous traitions, fonctionnaient de manière similaire en termes d'effacement, de déplacement ou de substitution, par exemple certaines propositions relatives, les appositions, les épithètes et les compléments de nom. Il nous est alors paru indispensable de disposer d'une grammaire qui soit constituée d'éléments factorisant la propriété syntaxique centrale à notre approche qui est celle d'effacement potentiel. La notion de constituant gouverné, définie dans GB, correspond assez fidèlement à cette propriété.

Cette information structurelle est étroitement liée à l'importance des constituants et n'a pas encore été utilisée de manière explicite et approfondie pour la production de résumé par compression structurelle. Nous définissons ici uniquement les grandes lignes de cette théorie, en nous concentrant sur les propriétés qui nous intéressent dans notre approche, et sans rentrer dans les nombreux cas particuliers, ni dans une formalisation poussée des concepts définis par N. Chomsky. Pour plus de détails, se référer aux ouvrages de référence cités en introduction de ce chapitre, ou à des ouvrages introductifs comme [Black, 1996, Schneider, 1998]. La théorie du gouvernement consiste en un ensemble de niveaux de représentations de la phrase : la structure profonde (*D-structure*), basée sur les relations fonctionnelles entre les différents éléments de la phrase, la structure de surface (*S-structure*), reflétant l'ordre de surface des éléments de la phrase, ainsi que les formes phonologiques et logiques. Dans ce travail, nous resterons entre les niveaux de la structure de surface et de la structure profonde, car ce sont les plus abordables sur le plan de l'analyse automatique.

Après avoir présenté les bases du gouvernement, section 3.2.1, nous discutons de la classification des constituants gouvernés, selon leur caractère d'effaçabilité, section 3.2.2.

3.2.1 Le gouvernement syntaxique.

3.2.1.1 Tête gouvernante et constituant gouverné.

Dans une phrase, l'existence de chaque constituant est légitimée par la présence d'une tête syntaxique, laquelle gouverne ce constituant. Une telle tête est, dans le cas général (défini par N. Chomsky), un mot appartenant à l'ensemble des catégories lexicales[22] suivantes : nom (N), verbe (V), adjectif (A) et préposition (P). Par exemple, dans le constituant nominal *le gros chat*, le mot *chat* est la tête de ce constituant, et gouverne le constituant adjectival réduit à l'adjectif *gros*, ainsi que le déterminant *le*.

Les conjonctions de subordinations (C) peuvent aussi être considérées comme des têtes,

[22]Catégorie lexicale est un anglicisme que nous employons dans le sens de catégorie grammaticale.

gouvernant la proposition subordonnée. Les autres catégories lexicales, déterminant (D), adverbe[23] (Adv), conjonction de coordination ($Conj$), *etc.* ne sont pas considérées comme des têtes. Pour les différencier graphiquement, un exposant 0 est ajouté aux têtes (par exemple N^0).

Cependant, pour le cas des constituants ayant pour fonction sujet et prédicat, leur gouvernance est plus particulière, car ils ne disposent pas de tête visible dans la phrase. Dans la théorie *X-barre*, théorie sur les catégories syntaxiques dans GB [Chomsky, 1970] nécessaire à la représentation graphique du gouvernement, une catégorie fonctionnelle[24] de flexion du verbe, notée I, qui vient de $Infl$ ou *Inflection* de l'anglais, est utilisée comme tête des constituants verbaux, et est définie comme le gouverneur des constituants sujet et prédicat. La tête fonctionnelle est alors notée I^0. Sa projection maximale[25] est étiquetée IP (pour *Inflexion Phrase*) et désigne le constituant de la proposition complète[26].

Ainsi, chaque constituant de la phrase est gouverné par une tête, syntaxique ou fonctionnelle. De plus, chaque constituant gouverné dispose d'une fonction vis-à-vis de cette tête, qui peut-être, toujours selon GB, spécifieur, complément ou adjoint.

Les arbres syntagmatiques dans GB se construisent selon un ensemble de règles de transformation hors contexte. Elles se situent à l'échelle du constituant et restent théoriquement génériques à n'importe quelle catégorie lexicale. Nous présentons maintenant les principales et nécessaires à notre travail.

Soit le système de règles structurelles X-barre suivant (3.1–3.4) :

$$XP \rightarrow \text{spécifieur } X' \tag{3.1}$$

$$X' \rightarrow X^0 \text{ compléments} \tag{3.2}$$

$$X' \rightarrow X' \text{ conjonction } X' \tag{3.3}$$

$$X' \rightarrow \text{adjoint } X' \tag{3.4}$$

XP signifie un constituant[27] portant la catégorie X de sa tête, nous parlerons alors de catégorie du constituant pour la désigner. Les X' sont des nœuds internes aux constituants[28].

La règle 3.1 place le spécifieur dans la structure, la 3.2 l'ensemble des compléments gouvernés par la tête X^0 (chaque complément ainsi que le spécifieur sont aussi analysables

[23]Nous verrons par la suite que nous pouvons considérer les adverbes comme des têtes.

[24]La catégorie fonctionnelle ne dispose pas d'entrée lexicale. Elle est artificiellement créée pour la représentation syntaxique.

[25]Le constituant complet dont il est la tête.

[26]Dans le cas où il y aurait une seule proposition, le constituant est la phrase complète.

[27]P pour le mot anglais *Phrase* qui désigne le constituant.

[28]Les notations peuvent varier d'un ouvrage à un autre. De manière générale X est une catégorie lexicale, X^0 est la tête d'un constituant, $X' = X^1$ et $X'' = X^2 = XP$.

en tant que XP), la 3.3 les couples de constituants coordonnés et enfin la 3.4 l'adjoint. À noter que l'adjoint peut parfois se placer à droite du X', par exemple dans *un chat tigré*, l'ordre des éléments de la conclusion de la règle n'a donc pas d'importance. S'il y a plus de deux éléments coordonnés ou plus d'un adjoint, il suffit d'appliquer respectivement la règle 3.3 ou la règle 3.4 récursivement.

La figure 3.1 représente un exemple d'arbre syntagmatique utilisant ce système de règles (excepté la coordination) sur la phrase de l'exemple 3.5.

Exemple 3.5 *Un chat tigré mange un rongeur sous une pluie diluvienne.*

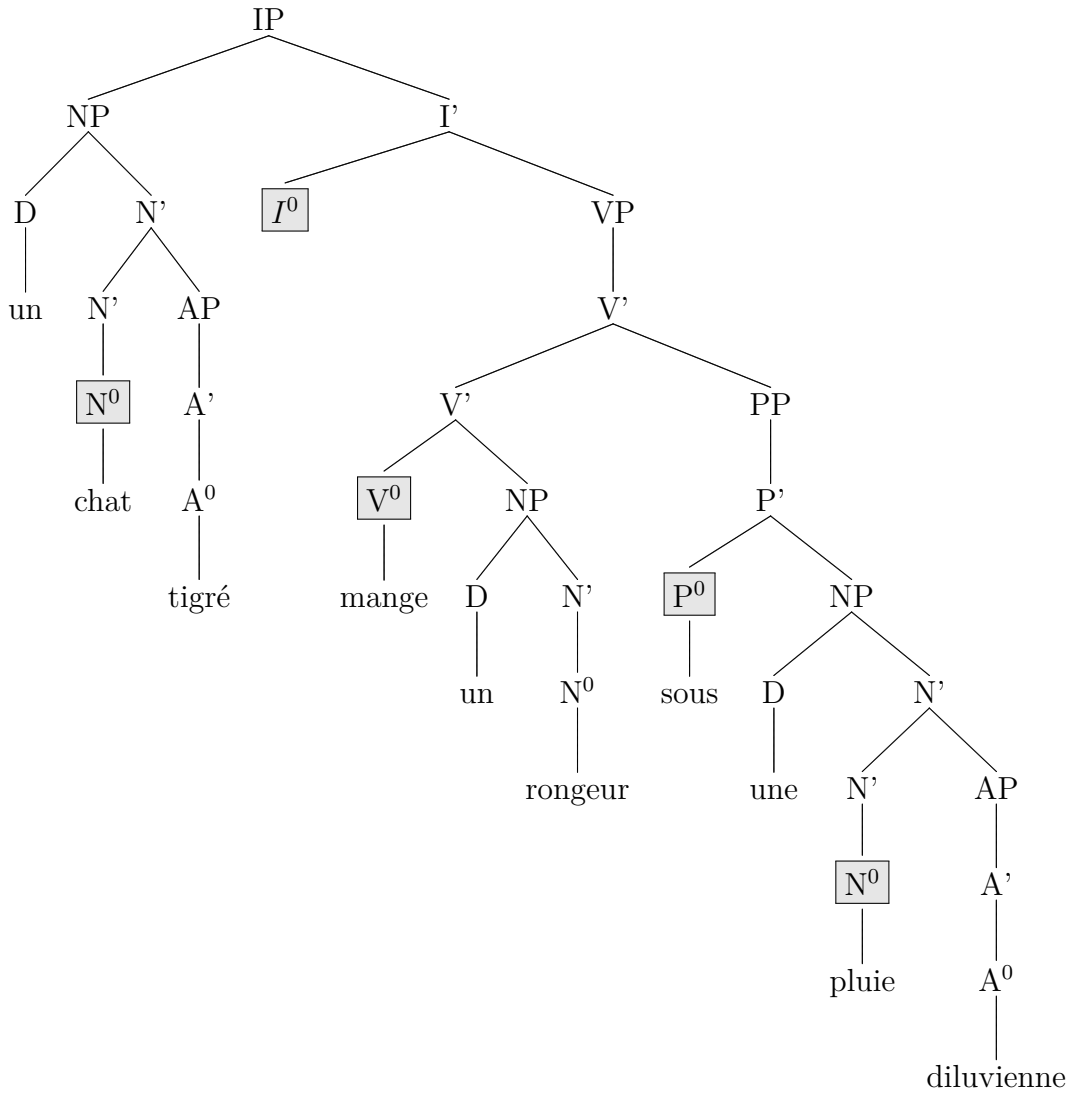

FIG. 3.1 – Exemple d'arbre syntagmatique selon la théorie X-barre.

En encadré et sur fond gris clair sont mis en avant les têtes gouvernantes qui nous intéressent dans cet exemple. Ainsi, I^0 gouverne le groupe sujet (son oncle syntaxique NP)

ainsi que le groupe prédicat (son frère syntaxique VP). D'après les règles structurelles, le groupe sujet est spécifieur de I^0, et le groupe prédicat son complément, nous reviendrons à cette vision particulière en section 3.2.2.5. De même, *chat* gouverne son déterminant *un* (D) et son adjectif *tigré* (AP), *mange* gouverne son complément *un rongeur* (NP) ainsi que son modifieur *sous une pluie diluvienne* (PP), enfin *sous* gouverne son complément *une pluie diluvienne*.

La relation de gouvernance nous intéresse pour deux principales raisons. La première est qu'une tête gouvernante, au sens de N. Chomsky, est considérée comme indispensable à la cohérence grammaticale et sémantique du constituant dans lequel elle est inscrite. En effet, la présence des constituants qu'elle gouverne n'est plus légitime si elle est supprimée, d'où l'incohérence résultante. La tête n'est donc pas effaçable de son constituant. La seconde raison est qu'un constituant gouverné est une unité syntaxique qui peut être **facultative sur le plan syntaxique** et **peu importante sur le plan sémantique**, ce constituant peut donc faire l'étude d'une suppression, sujet de la prochaine section.

3.2.2 La classification des éléments effaçables

Comme nous venons de le voir, N. Chomsky distingue, d'après sa définition dans GB, pour chaque catégorie lexicale, trois types de constituants gouvernés : les spécifieurs, les compléments et les adjoints. Nous discutons maintenant de cette classification selon nos attentes en termes de comportement sur l'effacement. Nous ne traitons pas de tous les cas particuliers de la langue, mais restons dans un cadre plus général, censé permettre le traitement de la majorité des cas syntaxiques à laquelle est confrontée une compression syntaxique de phrase. Les exemples seront majoritairement en français, le placement des constituants sera donc celui de la langue française, toutefois le principe reste général à la théorie X-barre.

Pour les fonctions qui entrent en conflit avec nos attentes, nous proposons des ajustements théoriques. Pour celles qui semblent en être absentes, nous proposons des ajouts théoriques. Nous nous appuyons parfois sur d'autres classifications existantes, issues d'autres grammaires, pour orienter nos propositions.

La première partie de cette section traite des spécifieurs, selon X-barre. La seconde et la troisième en font de même pour les compléments et les adjoints.

L'adverbe et le pronom ne sont généralement pas considérés en tant que tête dans les ouvrages traitant de la théorie X-barre. Nous abordons ces cas en section 3.2.2.4.

Notre point de vue sur la tête I^0 et ses constituants gouvernés diffère de celle de la théorie X-barre. Nous présentons notre adaptation en section 3.2.2.5.

La fonction de modifieur parfois rencontrée dans la littérature semble disposer de propriétés intéressantes pour notre approche. Nous en discutons en section 3.2.2.6.

Enfin nous résumons toutes ces modifications en section 3.2.2.7, en présentant notre classification finale.

3.2.2.1 Le spécifieur

Le spécifieur est défini principalement par rapport à sa position dans le constituant. En effet, la règle 3.1 le définit comme un élément situé complètement à gauche du constituant[29].

Ainsi, en anglais vont être considérés comme spécifieurs du nom les déterminants, comme *the cat*, mais aussi les cas possessifs : *the cat's food*. Cependant, en français, ce cas de possessif est exprimé par un complément du nom, situé à sa droite. En effet, en français, il n'y a pas de différence syntaxique marquée entre les compléments du nom (de la GC) *un chat de Marie* et *un chat de l'île de Man*, alors que dans la classification X-barre, en anglais, le premier fait partie d'un spécifieur et le second d'un complément. La fonction de l'élément syntaxique exprimant le cas du possessif est donc différente, au sens X-barre, entre le français et l'anglais. En termes d'effacement, le déterminant ne peut être supprimé, en français comme en anglais, alors que le cas possessif peut l'être. Il serait donc pratique d'attribuer au déterminant et au possessif des fonctions différentes.

Afin d'uniformiser le comportement du déterminant vis-à-vis de l'effacement, nous choisissons de le considérer comme un complément du nom plutôt que comme spécifieur. Cela est appuyé par le fait que les noms qui admettent un déterminant en requièrent obligatoirement sa présence. On retrouve alors la propriété de sous-catégorisation, propre au complément. Le déterminant devient donc un complément obligatoire du nom. La figure 3.2, basée sur le constituant *un chat tigré* de l'exemple 3.5, illustre la nouvelle représentation syntaxique du déterminant dans le constituant nominal. Le déterminant se plaçant à gauche du nom, nous considérons la règle 3.2 comme admettant des compléments de part et d'autre de la tête.

Dans nos arbres syntagmatiques, un triangle constitue un raccourci de représentation pour un sous-arbre dont le développement n'est pas pertinent dans l'illustration.

Le spécifieur de la préposition peut être un adverbe, effaçable, placé avant la préposition. Par exemple *Il arrive peu avant le début du cours.*

Les spécifieurs de l'adjectif et du verbe sont généralement des adverbes, effaçables, placés avant ou après selon la tête. Ceux-ci sont toujours effaçables.

[29] Il est cependant possible de modifier la règle afin de placer le spécifieur droite du constituant, si la langue analysée le requiert.

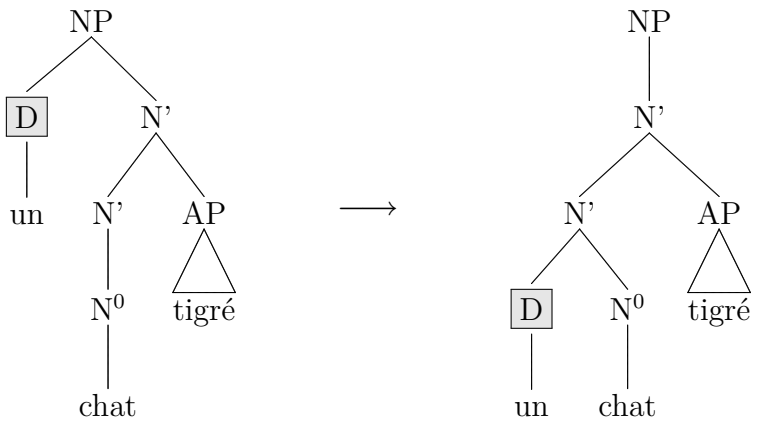

Fig. 3.2 – Exemple de déterminant comme complément.

3.2.2.2 Le complément

Le complément est un constituant sous-catégorisé par sa tête, qui lui est généralement contiguë. Dans l'arbre syntagmatique, la règle 3.2 le place comme frère de la tête.

En voici quelques exemples : pour le nom, *un chat de l'île de Man* ; pour l'adjectif, *tigré de la tête* ; pour la préposition, *de la tête* et pour le verbe, *mange une souris*. Le complément du nom est généralement effaçable, mais ce n'est pas toujours le cas pour le verbe et l'adjectif, et jamais le cas pour préposition.

Pour le complément du verbe, certains verbes imposent la présence d'un ou deux compléments, comme le verbe *donner* : *Jean donne une croquette à Félix*. Dans ce cas, les compléments ne sont pas ou difficilement effaçables. S'ils sont effacés, la phrase devient agrammaticale, ou est transformée en une formulation peu courante, dont le sens est généralement modifié. Ce caractère de non-effaçabilité dépend des traits lexicaux de chaque verbe, plutôt que d'une règle générale, et devra donc être traité au cas par cas.

Un cas particulier pour les compléments du verbe sont les attributs du sujet de la grammaire classique. Leur construction dans la phrase les identifie en tant que complément du verbe être, d'après les règles de la théorie X-barre. Leur fonction syntaxique peut donc être considérée comme complément du verbe. Cependant, leur fonction sémantique est différente, car ils sont attributs du sujet. Ils sont d'ailleurs indispensables à la cohérence syntaxique de la phrase. Ce type de constituant gouverné ne pose cependant pas de problème à notre modèle basé sur l'effaçabilité, du moment que les verbes d'états sont considérés comme des verbes classiques admettant un complément obligatoire de type groupe adjectival.

Le cas des attributs du complément de l'objet de la grammaire classique peut être traité de manière similaire. Les verbes autorisant de tels compléments ainsi que leur attri-

buts peuvent être considérés comme admettant deux compléments obligatoires, un groupe nominal et un groupe adjectival. Ceci concorde avec le principe général du gouvernement, car l'existence d'un tel attribut dépend de celle du verbe. Sans le verbe, la présence de cet attribut n'est pas légitime.

L'effaçabilité des compléments de l'adjectif dépend de leur tête. Maurice Grevisse écrit [Grevisse, 1993–1997], au sujet des adjectifs : « les compléments sont essentiels, 1° soit quand leur construction [...] dépend de l'adjectif support des compléments ; 2° soit quand l'adjectif ne peut s'employer sans eux ; — les compléments sont non essentiels quand ils n'obéissent à aucune de ces deux conditions. » Une étude au cas par cas est donc nécessaire. Par exemple, les adjectifs suivant ne semblent pas s'employer sans complément : *natif de quelque part*, *enclin à faire quelque chose*. Par contre les adjectifs *prêt* et *content* peuvent se passer de complément.

Enfin, concernant le complément de la préposition, il est toujours indispensable à la cohérence syntaxique et sémantique. Dans l'exemple 3.5, le constituant *une pluie diluvienne* est complément de la préposition *sous*.

Nous parcourons maintenant plusieurs fonctions qui trouvent des intersections avec le complément défini selon la propriété de sous-catégorisation. Cette étude s'appuie sur le rapport interne [Mela, 2007].

L'argument. Une fonction proche de celle du complément est l'*argument*, lequel concerne un constituant généralement attaché à un verbe, dans une relation de prédicat – argument. Notons que ce terme est utilisé par N. Chomsky dans sa θ-théorie, [Chomsky, 1981, Chomsky, 1986], pour désigner les constituants sélectionnés par les *prédicats*, appelés *catégories-opérateurs*. Cette relation est de nature sémantico-logique, de type agent, patient, but, thème, source, *etc.* et peut s'appliquer à une tête de n'importe quelle catégorie lexicale comme catégorie-opérateur. Par exemple, dans la phrase *Félix chasse des souris pour se divertir*, le prédicat-opérateur est le verbe *chasse*, l'agent est le nom *Félix*, le patient est le groupe nominal *des souris* et le but est le groupe prépositionnel *pour se divertir*. Notre approche se concentrant sur les informations syntaxiques de la phrase, plutôt que sur les informations sémantiques ou logiques, nous n'étudierons pas davantage cette information.

Le complément dans le $TLFi$. Il est définit comme un « mot ou groupe de mots de nature substantivale mis en relation de subordination immédiate avec une unité signifiante pour en compléter ou en préciser le sens ». La limitation à la nature substantivale proscrit de l'ensemble de leurs compléments les cas de constituants possédant les natures de propositions subordonnées, d'adverbes, d'adjectifs, d'adverbes et de pronoms. Cette définition ne peut donc pas s'appliquer à l'ensemble des constituants effaçables qui nous

intéresse. Elle n'aborde pas non plus le trait d'effaçabilité qui nous est important.

Le complément dans le Bescherelle. Le Bescherelle de la grammaire, [Bes1990], aborde le complément sous plusieurs points de vue, certains issus de la GC, d'autres de différentes grammaires plus modernes dont certaines conviennent davantage à notre approche.

Les compléments de la GC décrits dans l'ouvrage sont ceux de l'adjectif, d'agent, d'attribution, circonstanciel, du nom et d'objet (direct, indirect et second). Ces compléments ne sont pas adéquats à notre approche car leur définition varie selon les cas, il n'y a pas d'uniformité dans le comportement fonctionnel. Voici plusieurs exemples :

- le complément circonstanciel est parfois adjoint au verbe (*Je travaille à Paris*) ou sous-catégorisé par le verbe (exemple 3.6) ;
- le complément de l'adverbe et du verbe ne sont pas définis[30] ;
- le complément de l'adjectif et celui d'agent sont tous deux sous-catégorisés par un adjectif ;
- l'épithète et la proposition subordonnée relative sont tous deux des adjoints du nom[31].

Exemple 3.6 *Je vais à Paris.*

Cet ouvrage aborde aussi les compléments du verbe sous un autre point de vue, plus récent que celui de la GC qui les sépare en deux classes (page 71) :

- les compléments essentiels ;
- les compléments circonstanciels.

Leur définition est basée sur deux critères qui sont ceux d'effaçabilité et de déplaçabilité : le complément essentiel n'est ni effaçable, ni déplaçable, alors que le circonstanciel est les deux à la fois. Les limites de cette classification sont atteintes pour les compléments qui sont ni effaçables, ni déplaçables mais expriment toutefois les circonstances de l'action. Par exemple dans l'exemple 3.6.

Enfin, le Bescherelle aborde une dernière classification au sujet des compléments (page 73) :

- les compléments du verbe ;
- les compléments de la phrase.

[30] Les compléments d'objets sont des compléments du verbe, mais ne sont pas explicitement définis comme tels. De plus sont manquants les compléments du verbe de type circonstanciels (exemple 3.6).

[31] En page 241 on peut lire que la subordonnée relative « fait partie des expansions du nom au même titre que le complément du nom et l'adjectif ». On retrouve bien une notion de gouvernement, mais qui n'est présentée que comme un trait secondaire.

Le but de cette classification est de séparer les compléments qui ne modifient que les verbes de ceux qui apportent une information à la phrase complète. Ces compléments du verbe correspondent fidèlement à ceux définis dans GB. Par contre, ces compléments de la phrase ne correspondent pas à des compléments dans GB, mais plutôt aux adjoints verbaux[32]. Le fait de les attacher à la phrase plutôt qu'au verbe a retenu notre attention, nous développons ce point en section 3.2.2.5.

3.2.2.3 L'adjoint

L'adjoint peut se définir comme un constituant optionnel qui n'est ni un complément, ni un spécifieur. La règle 3.4 place l'adjoint en frère d'un X', donc soit en oncle, soit en grand-oncle, *etc.* de la tête.

Pour le nom, ce sont généralement les constituants tenant lieu de groupes adjectivaux qui sont adjoints, par exemple dans *un chat tigré*, mais d'autres catégories de constituants sont possibles : les subordonnées relatives et complétives, les groupes nominaux et participiaux.

Pour l'adjectif, ils sont assez rares, ils peuvent se réaliser par exemple en groupes adverbiaux, comme dans *tigré comme peu de chats*.

Pour la préposition, peu fréquents aussi, en groupe prépositionnel par exemple : *Marie est en colère contre son chat*.

Enfin pour le verbe, les adjoints de la théorie X-barre correspondent en général aux traditionnels compléments circonstanciels[33], comme *sous une pluie diluvienne* de l'exemple 3.5.

Cette fonction d'adjonction correspond à nos attentes en termes de régularité vis-à-vis de l'effacement.

3.2.2.4 L'adverbe et le pronom en tant que têtes syntaxiques

L'adverbe en tant que tête. L'adverbe n'est généralement pas présenté comme une tête syntaxique dans les ouvrages traitant de GB, cependant il peut gouverner certains constituants, selon la définition du gouvernement, et la généricité des règles de la théorie X-barre n'interdit pas qu'il soit en position de tête, comme nous allons le voir maintenant.

Comme spécifieur de l'adverbe, peut se trouver un autre adverbe, comme dans *vraiment indépendamment*, de l'exemple 3.7. Un complément de l'adverbe peut aussi se réaliser, comme dans *de la volonté de son chat*, du même exemple.

[32]Nous pouvons aussi noter que ces compléments de la phrase ne sont pas sous-catégorisés par le verbe.
[33]Ceux qui ne sont pas compléments du verbe.

Exemple 3.7 *Marie agit vraiment indépendamment de la volonté de son chat.*

L'arbre syntagmatique de la figure 3.3 est la représentation en X-barre de l'exemple 3.7, la tête du constituant adverbial (Adv^0), ainsi que son complément (PP) et son modifieur ($AdvP$) sont encadrés.

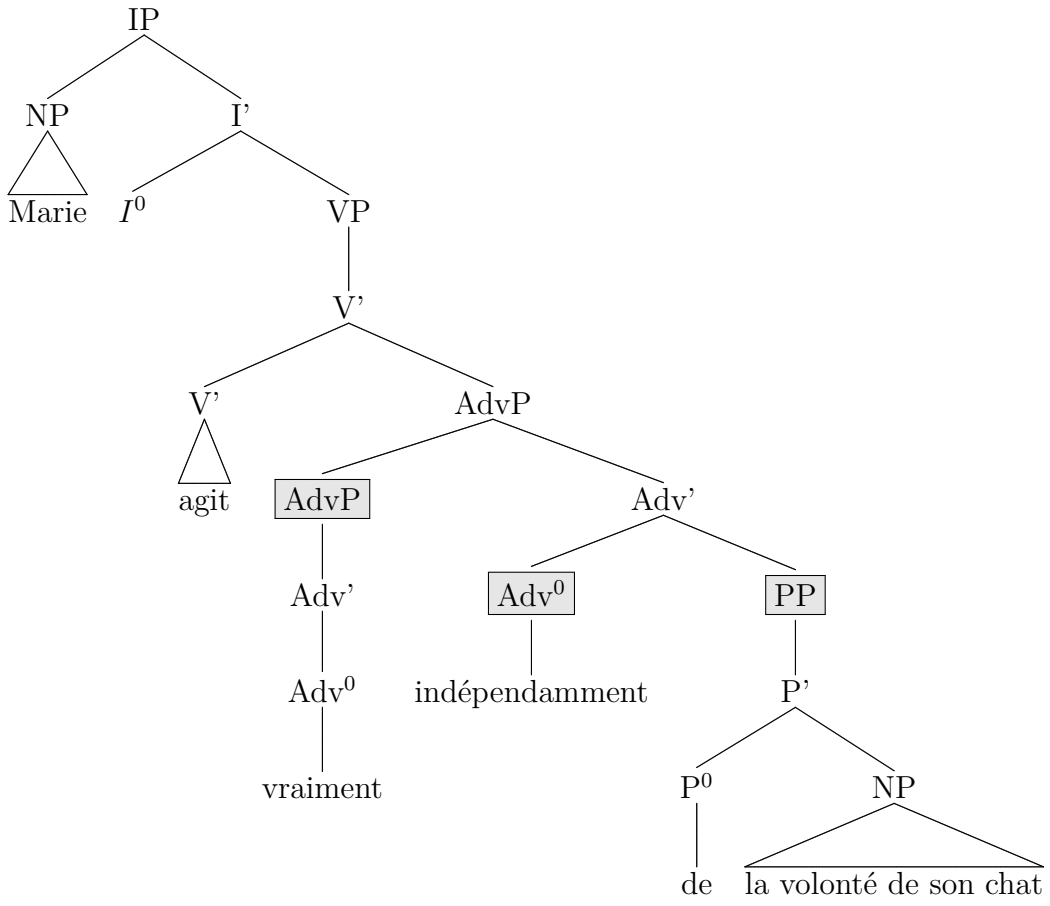

FIG. 3.3 – Exemple d'arbre syntagmatique incluant un constituant adverbial.

L'effaçabilité des compléments de l'adverbe dépend directement de l'adverbe lui-même. Certains adverbes (voir [Grevisse, 1993–1997], pages 596–597) nécessitent systématiquement un complément de type :

- groupe prépositionnel, comme *antérieurement*, *conformément*, *conséquemment*, *parallèlement* ;
- proposition conjonctive corrélative[34], comme *plus*, *davantage*, *moins*, *autant*, *aussi* ;
- proposition relative (pour certains adverbes de lieu et de temps), comme *partout*, *maintenant*.

[34]C'est le cas des comparatifs. La proposition appelée par ces adverbes peut alors être considérée comme un complément de l'adverbe, car sous-catégorisée par ce dernier, malgré le fait qu'elle en soit généralement séparée par la tête de l'adverbe (adjectif). Nous verrons à ce sujet, section 3.2.2.7 que la théorie GB peut accepter de telles irrégularités.

D'autres n'en nécessitent pas forcément un, comme ceux exprimant le degré : *je mange beaucoup de légumes*, mais aussi *je mange beaucoup*.

Le pronom en tant que tête. Tout comme l'adverbe, le pronom peut être traité comme une tête syntaxique.

Dans certains cas, il peut fonctionner de manière similaire au nom :

- sans déterminant : *Il mange* ;
- avec déterminant : *Le mien mange.*

Au niveau des constituants gouvernés par le pronom, une différence notable se fait, par rapport au nom, pour les compléments, lesquels sont parfois obligatoires :

- *Celui qui abîme le canapé aura affaire à Marie.*
- *[35] *Celui* ⊗ [36] *aura affaire à Marie.*
- *Tous les chats, même ceux préférés de Jean, auront affaire à Marie.*
- * *Tous les chats, même ceux* ⊗ *, auront affaire à Marie.*
- *Lequel des chats aura le poulet ?*
- *Lequel* ⊗ *aura le poulet ?*

C'est généralement le type du pronom qui détermine le caractère obligatoire du complément. Le pronom *celui* (ainsi que ses différentes flexions), *ce*, *un* et *une* semble requérir systématiquement un complément.

Certains pronoms peuvent parfois admettre un spécifieur. Par exemple, pour le pronom *tout*, dans la phrase suivante : *Les filles sont presque toutes rentrées.*

3.2.2.5 Les constituants gouvernés par I^0

Spécifieur et compléments de I^0. Comme nous l'avons vu en section 3.2.1, le groupe sujet est considéré, dans GB, comme spécifieur de I^0, et le groupe prédicat comme complément de cette tête fonctionnelle. Cependant, si nous considérons la propriété de sous-catégorisation, propre aux compléments, le sujet la vérifie. La tête I^0 requiert bien la présence du groupe verbal[37], et mais aussi celle du groupe nominal (le sujet), excepté pour les formes impératives. Nous pouvons alors considérer que le sujet est aussi complément de I^0. Le caractère de non-effaçabilité de ces deux constituants en fait des compléments obligatoires de la tête I^0. La construction syntaxique classique de ces deux compléments est sujet puis prédicat[38].

[35] Une phrase agrammaticale est, dans ce travail comme généralement dans la littérature, désignée par un astérisque la précédant.

[36] Le symbole « ⊗ » est utilisé dans ce travail pour marquer la suppression d'un élément de la phrase. Cette dernière ne cause cependant pas systématiquement une agrammaticalité de la phrase.

[37] Nous ne considérons pas ici les rares cas de phrases nominales, comme pour les titres, pour lesquels leur construction fait office d'exception et leur compression n'est pas pertinente.

[38] Les formes interrogatives constituent des modifications de la structure profonde.

L'arbre syntagmatique de la figure 3.4 illustre cette nouvelle considération d'attachement.

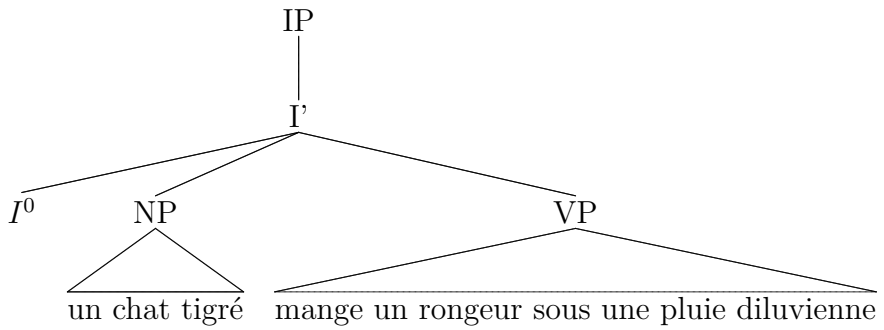

FIG. 3.4 – Exemple de groupe sujet et de groupe prédicat compléments de I^0.

La position de la tête I^0 tout à gauche des compléments est arbitraire, car elle ne dispose pas de réalisation lexicale. Cette adaptation syntaxique nous permet de justifier, dans notre classification, le caractère ineffaçable du sujet.

Adjoints de I^0. Comme nous l'avons vu en section 3.2.2.2, l'attachement des adjoints au verbe peut être remis en question. Tout d'abord, ces adjoints semblent modifier la proposition[39] plutôt que le verbe. Sachant que nous ne considérons pas les relations d'un point de vue sémantico-logique, comme c'est le cas pour la θ-théorie vue précédemment, nous détachons alors le groupe verbal du groupe prédicat, les considérant séparément. De ce point de vue, un constituant qui modifie le couple (sujet, prédicat) ne peut donc être considéré comme un adjoint du verbe seul.

De plus, ces adjoints verbaux ont la faculté de pouvoir être déplacés en de nombreuses positions dans la proposition, certaines suffisamment éloignées pour remettre en cause la gouvernance de la tête verbale, comme dans les phrases de l'exemple 3.8, basées sur l'exemple 3.5.

Exemple 3.8

a *Sous une pluie diluvienne, un chat tigré mange un rongeur.*

b *Un chat tigré, sous une pluie diluvienne, mange un rongeur.*

c *Un chat tigré mange, sous une pluie diluvienne, un rongeur.*

d *Un chat tigré mange un rongeur sous une pluie diluvienne.*

[39]Nous parlons ici de proposition plutôt que de phrase, car telle est la limite de portée de ces constituants. Par exemple, dans la phrase suivante, les deux constituants de ce type ne modifient que la proposition dans laquelle ils sont inscrits : *Hier soir, Félix, qui jouait dans la maison, se blessa.* Le premier précise quand Félix s'est blessé, alors que le second précise où Félix jouait.

En effet, les règles structurelles X-barre (3.1–3.4) ne permettent pas aux adjoints d'être séparés de leur tête par un constituant non gouverné par cette dernière, comme dans la phrase a, ni d'être placés entre leur tête et un complément, comme dans la phrase c.

GB autorise toutefois certaines irrégularités de placement, en les considérant comme des déplacements par rapport à la structure profonde de la phrase, et en autorisant des manipulations structurelles consistant à la simple application d'une règle de déplacement dans l'arbre syntagmatique, appelée $Move - \alpha$, pour réobtenir cette structure profonde, laquelle vérifie alors les règles structurelles de la théorie X-barre. Ainsi, dans la phrase a ou c de l'exemple 3.8, il suffit d'appliquer une règle de déplacement sur le constituant *sous une pluie diluvienne*, pour le restituer à une place d'adjoint du verbe autorisée par le système de règles de base de la théorie X-barre, c'est-à-dire comme dans les phrases b ou d.

Le choix de l'attachement de ce constituant à la phrase ou au verbe semble être une question de point de vue, et de propriétés considérées. Dans [Grevisse, 1993–1997], page 500, l'auteur confirme cette divergence d'opinion et écrit, en parlant de ces constituants : « Étant donné la mobilité toute particulière [...] de la plupart des compléments non essentiels, certains grammairiens refusent de les ranger parmi les compléments du verbe et parlent à ce sujet de *complément de phrase* ».

Le but de cette section est d'établir une classification des constituants supprimables selon leur fonction syntaxique, donc selon leur attachement dans la phrase. Ces adjoints verbaux semblent davantage modifier la proposition que le verbe, nous les considérons donc comme attachés à la proposition et leur attribuons la fonction de modifieur de la proposition. D'un point de vue de la syntaxe X-barre, en prenant aussi en compte notre modification sur la gouvernance de I^0 vue au paragraphe précédent, cet attachement peut être considéré comme celui d'une adjonction à la tête fonctionnelle I^0. Les positions qui ne respectent pas cet attachement (comme dans les phrases b et c) peuvent s'expliquer en termes d'opérations $Move - \alpha$.

Les arbres des figures 3.5 et 3.6 illustrent un tel attachement à I^0 pour les phrases a et d, respectivement.

Nous verrons dans le chapitre suivant, que l'outil d'analyse syntaxique utilisé, a fait un choix similaire sur l'attachement.

3.2.2.6 Le modifieur

Le terme de *modifieur* (ou modificateur) est issu du mot anglais *modifier*, utilisé typiquement dans la grammaire anglaise. Nous appelons modifieur un constituant qui restreint ou qualifie sémantiquement un autre constituant et qui est facultatif sur le plan syntaxique . Il semble donc correspondre au spécifieur dans GB, si on ne tient pas compte des quelques

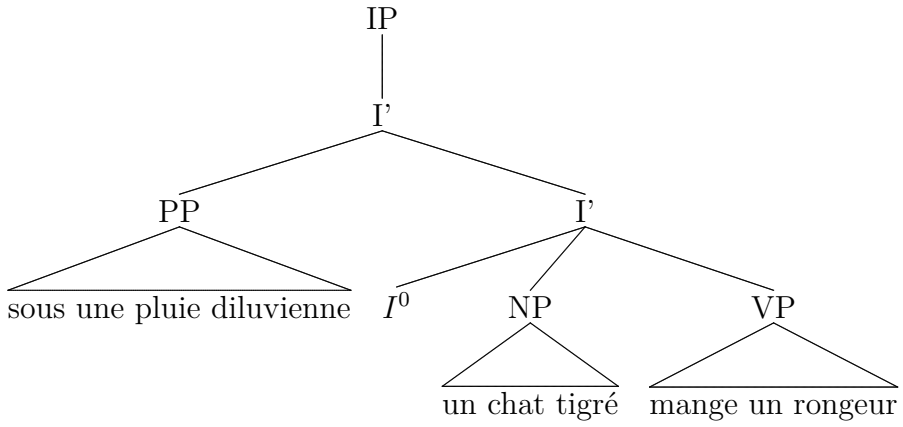

FIG. 3.5 – Exemple d'adjoint de la proposition, placé à sa gauche.

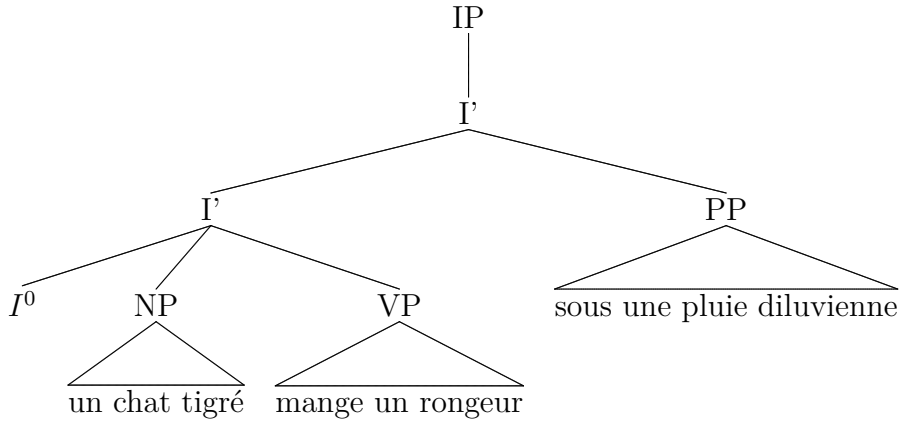

FIG. 3.6 – Exemple d'adjoint de la proposition, placé à sa droite.

exceptions citées en section 3.2.2.1 (pour les déterminants et les groupes sujets).

Le modifieur est aussi repris dans certains travaux traitant de grammaire française et a notamment été utilisé dans la campagne d'évaluation EASY[40] (Évaluation des Analyseurs SYntaxiques), présentée à TALN & RECITAL 2005 à Dourdan. Dans cette campagne, le principe de modifieur n'est pas défini dans sans globalité mais plutôt au cas par cas et désigne un constituant qui modifie un autre. Le guide d'annotation[41] regroupe les différents cas de modifieurs considérés pour la campagne : les modifieurs du verbe, du nom, de l'adjectif, de l'adverbe et de la préposition. La fonction attribuée à ces modifieurs regroupe celles des spécifieur et d'adjoint dans GB, encore une fois à l'exception des cas particuliers cités en section 3.2.2.1. Notre modifieur de la proposition correspond à certains de leurs modifieurs du verbe, lesquels mélangent des modifications du prédicat comme dans le constituant *Félix dort profondément*, avec des modifications que nous considérons

[40]Site Internet : `http://www.limsi.fr/Recherche/CORVAL/easy/`

[41]Page Internet : `http://www.limsi.fr/Recherche/CORVAL/easy/PEAS_reference_annotations_v1.6.html`

effectuées sur la proposition, comme dans le constituant *Félix dort à toute heure*.

Cette classification voit un avantage certain à notre approche : les constituants systématiquement facultatifs sur le plan syntaxique et non sous-catégorisés par leur tête se voient tous attribuer la même fonction, celle de modifieur.

3.2.2.7 Notre classification

Le modifieur. Nous avons retenu le terme de modifieur pour regrouper les fonctions de spécifieur et adjoint de GB, en y incluant les modifieurs de la proposition, mais en y excluant les déterminants et les groupes sujets. Nous caractérisons donc notre première classe de constituants gouvernés, les modifieurs, par les propriétés de :

1. modification ou de précision du sens de la tête ;
2. d'effacement toujours possible ;
3. de non sous-catégorisation par la tête.

Le complément. Le caractère d'effacement des compléments dans GB dépend des traits lexicaux de leur tête. Nous caractérisons alors notre deuxième classe de constituants gouvernés, les compléments, par la propriété de sous-catégorisation par leur tête, laquelle défini la possibilité d'effacement du complément. Cette classe correspond assez étroitement à celle des compléments dans GB. Nous y ajoutons cependant le déterminant en tant que complément obligatoire du nom (commun), ainsi que les groupes sujets et prédicats en tant que compléments obligatoires de la proposition.

Les autres têtes lexicales. Notre classification inclut aussi les têtes lexicales adverbiales et pronominales.

Le système de règles structurelles adapté. Enfin, le système de règles structurelles X-barre (3.1–3.4), peut être adapté à nos attachements syntaxiques. Soit le système de règles structurelles X-barre adapté à notre approche (3.5–3.9) suivant :

$$XP \rightarrow X' \tag{3.5}$$

$$X' \rightarrow \text{compléments } X^0 \text{ compléments} \tag{3.6}$$

$$X' \rightarrow X' \text{ conjonction } X' \tag{3.7}$$

$$X' \rightarrow \text{modifieur } X' \tag{3.8}$$

$$X' \rightarrow X' \text{ modifieur} \tag{3.9}$$

La règle 3.5 permet de générer la racine du constituant, garantissant ainsi l'unicité du nœud XP pour chaque constituant. Elle remplace l'ancienne règle 3.1, après l'élimination du spécifieur. La règle 3.6, remplaçant la règle 3.2, généralise le placement des compléments de part et d'autre de leur tête. Ainsi, pour le nom, le déterminant peut être placé par une application directe de cette règle, tout en considérant un éventuel autre complément du nom placé à sa droite. La règle 3.7 est identique à la règle 3.7. Enfin les règles 3.8 et 3.9, remplaçant la règle 3.4, autorisent le placement de chaque modifieur à gauche ou à droite de la tête et ses compléments.

La principale exception à ces règles concerne les modifieurs de la proposition, qui peuvent admettre davantage de positions, lesquelles peuvent s'expliquer en termes d'opérations $Move-\alpha$ sur la structure de surface, pour rétablir l'ordre de la structure profonde, lequel vérifie alors le système de règles.

Récapitulatif. Nous obtenons donc deux classes de constituants syntaxiques potentiellement effaçables, les modifieurs et les compléments, lesquels peuvent être gouvernés par le nom, le pronom, l'adjectif, le verbe, l'adverbe, la préposition ou la proposition. Le tableau 3.1 est un récapitulatif des possibilités d'effacement des modifieurs et complément.

	nom	pron.	adj.	verbe	adverbe	prép.	proposition
modifieur	✓	✓	✓	✓	✓	✓	✓
complément	✓/✗	✓/✗	✓/✗	✓/✗	✓/✗	✗	✗

Tab. 3.1 – L'effacement des modifieurs et compléments.

Le symbole ✓ signifie que le constituant est toujours supprimable, le symbole ✗ qu'il ne l'est jamais et le couple ✓/✗ que son effacement dépend de sa tête.

Illustration de notre compression basée sur les modifieurs et compléments. Reprenons l'exemple 3.5, et procédons à une compression syntaxique manuelle, exemple 3.9, en traitant le cas de chaque modifieur et complément. Le nouvel arbre syntagmatique de cette phrase, basé sur notre classification, est représenté en figure 3.7

À l'échelle de la proposition, pour le maintien de la cohérence grammaticale, on ne peut supprimer le sujet (a), ni le verbe (b), compléments obligatoires de la proposition. Par contre, le modifieur *sous une pluie diluvienne*, peut être supprimé sans nuire à la cohérence syntaxique et de plus sans perte importante de contenu informationnel (c).

À l'échelle du sujet, le groupe nominal peut être réduit en supprimant le modifieur *tigré* (d). Par contre, la suppression du déterminant *un*, complément obligatoire, cause une incohérence syntaxique (e). Le cas de la suppression de la tête (f) est particulier, car l'adjectif est substantivable, prenant alors la place du nom s'il est effacé. Si *tigré* est pris

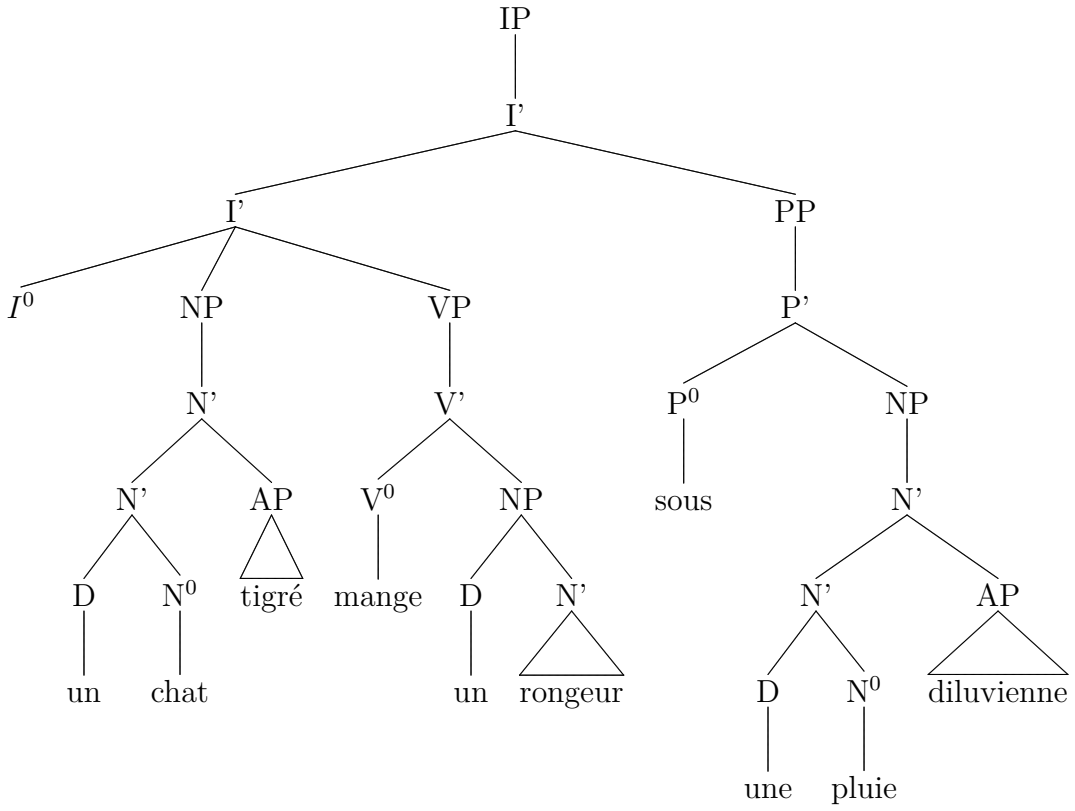

FIG. 3.7 – Exemple d'arbre syntagmatique selon nos règles structurelles (3.5–3.9).

comme adjectif, alors un déterminant suivi d'un adjectif est agrammatical. Cependant, l'interprétation commune ici sera celle d'un nom, rendant la phrase grammaticale. Ce genre d'exceptions sort du cadre de notre étude générale, dans lequel nous avons fait le choix de préserver les fonctions syntaxiques de tous les constituants de la phrase dans la compression produite. Nous ne les traiterons donc pas davantage.

À l'échelle du groupe verbal, la tête (le verbe) ne peut naturellement pas être supprimée (g). Si nous pénétrons dans le groupe prépositionnel, d'une part nous retrouvons les cas du déterminant, de la tête et du modifieur du nom, dont le traitement aboutit aux mêmes résultats que pour le groupe sujet, et d'autre part nous pouvons observer la vérification de la propriété de non-suppression du complément de la préposition (h). Enfin, le complément du verbe *un rongeur* peut être supprimé (i), car le verbe *manger* admet une forme intransitive, dont le sens est identique à la forme transitive de l'exemple. Nous pouvons éventuellement noter une plus grande perte d'information que lors de la suppression du modifieur, cependant la phrase reste grammaticale et l'information la plus centrale, c'est-à-dire le fait qu'un chat soit en train de se nourrir, est conservée.

L'application de toutes ces suppressions, préservant la grammaticalité, est représentée en j.

Exemple 3.9

a * ⊗ *mange un rongeur sous une pluie diluvienne*[42].

b * *Un chat tigré*[43] ⊗ .

c *Un chat tigré mange un rongeur* ⊗ .

d *Un chat* ⊗ *mange un rongeur sous une pluie diluvienne.*

e * ⊗ *chat tigré mange un rongeur sous une pluie diluvienne.*

f (*)[44] *Un* ⊗ *tigré mange un rongeur sous une pluie diluvienne.*

g * *Un chat* ⊗ *un rongeur sous une pluie diluvienne.*

h * *Un chat mange un rongeur sous* ⊗ .

i *Un chat tigré mange* ⊗ *sous une pluie diluvienne.*

j *Un chat* ⊗ *mange* ⊗ ⊗ .

Conclusion. Dans cette section, nous avons étudié l'effaçabilité des constituants selon leur fonction syntaxique, définie principalement à partir de la théorie X-barre. La prochaine section consiste à poursuivre notre étude d'importance des constituants, mais en explorant d'autres traits linguistiques. Ces nouvelles informations nous permettront à la fois de trancher sur l'effaçabilité de certains compléments, mais aussi d'améliorer le choix de suppression de nos modifieurs et compléments selon leur importance.

3.3 Exploitation de traits linguistiques dans notre compression de phrases

Dans cette section, nous décrivons un ensemble de méthodes d'exploitation de traits linguistiques des éléments de la phrase, dans le but d'améliorer notre compression syntaxique. Nous portons également notre attention sur les informations nécessaires aux ressources linguistiques utilisées, ainsi qu'aux conséquences de manques ou d'approximations de ces informations. La première partie (section 3.3.1) se concentre sur l'information de sous-catégorisation, donc sur les compléments. La seconde (section 3.3.2) explore les relations de fonction lexicales impliquées entre les têtes et leurs constituants gouvernés.

[42]Cette structure peut être validée par une construction impérative, cependant la construction syntaxique est différente et le sens véhiculé a radicalement changé. Nous considérons alors ce genre de cas similaire à ceux d'agrammaticalité. Une telle compression est une très mauvaise représentante de l'originale.

[43]Encore une fois, cette construction peut être acceptée, ici comme phrase nominale. Cependant, pour la même raison que dans la précédente phrase, elle ne peut constituer une bonne compression.

[44]Comme expliqué précédemment, la phrase résultante n'est pas agrammaticale, mais le mot *tigré* a changé de catégorie lexicale.

Enfin, la dernière partie (section 3.3.3) traite d'autres traits linguistiques exploitables dans notre cadre.

3.3.1 Exploitation de la sous-catégorisation

Nous décrivons à présent une méthode qui exploite des informations de sous-catégorisation dans le but d'améliorer la finesse d'analyse de l'importance des compléments.

Comme nous l'avons vu dans le chapitre précédent, [Jing, 2000] exploite une telle information au sujet des compléments du verbe afin de déterminer s'ils sont obligatoires. Les auteurs utilisent cette information pour partiellement palier les problèmes, inévitables à l'heure actuelle, d'analyse syntaxique limitée, principalement pour les erreurs d'attachement des groupes prépositionnels aux verbes. Notre but est de reprendre ce principe, tout en l'étendant :

- à davantage de nuances sur l'importance des compléments et
- à davantage de catégories lexicales pour les têtes concernées.

En effet, comme nous l'avons identifié dans la précédente section, l'adjectif, l'adverbe et le pronom déterminent tout comme le verbe la possibilité d'effacement de leur complément.

Afin de décrire notre méthode d'exploitation du trait de sous-catégorisation, nous utilisons, comme support, une ressource lexicale adaptée au traitement automatique. Elle nous permet de décrire notre méthode de telle sorte qu'elle puisse être mise en œuvre sur une telle ressource. Cette section débute par une courte introduction (section 3.3.1.1) au contexte linguistique introduisant cette ressource, puis nous décrivons la représentation des compléments dans la ressource (section 3.3.1.2), enfin nous décrivons comment en exploiter ces informations de sous-catégorisation pour déterminer l'effaçabilité des compléments (section 3.3.1.3).

3.3.1.1 Des grammaires universelles aux ressources lexicales

Le développement des récentes ressources lexicales, potentiellement candidates à notre approche, s'appuie sur les grammaires d'unification. Le domaine s'est constitué un ensemble de telles grammaires, développées en marge du modèle syntaxique *chomskyen*, qui ont pour but d'unifier l'étude de la syntaxe avec celle du lexique et de la sémantique, et de permettre, par une formalisation plus explicite, une facilitation des applications en traitement du langage. Parmi les plus connues de ces grammaires, on trouve HPSG [Pollard & Sag, 1994] et LFG [Kaplan & Bresnan, 1982], qui s'appuient sur la réécriture de clauses, complétée par l'emploi de l'unification pour modéliser des informations partielles (sous-spécification), ou les Grammaires d'Arbres Adjoints (*Tree Adjunct Grammar*, TAG) [Joshi, 1987] et ses nombreuses variantes qui reposent sur la combinaison de structures grammaticales élémentaires, en l'occurrence des arbres, à l'aide de divers opérateurs

de dérivation, comme l'adjonction. Ces grammaires visent, entre autres, à décrire la phrase non seulement au niveau de ses dépendances syntaxiques, mais aussi au niveau des dépendances lexicales, représentées par une structure de traits lexicaux. C'est cette structure qui nous intéresse, car elle permet de nous renseigner, pour chaque tête ou classe particulière de têtes, sur leurs différentes constructions syntaxiques, notamment les compléments optionnels et obligatoires.

Une des ressources parmi les plus connues et les plus complètes, pour le français, exploitant une telle structure, est le lexique-grammaire. Ce lexique recense, en grandeur réelle, les structures syntaxiques élémentaires [Gross, 1975]. Son développement a été commencé par Maurice Gross au LADL[45] et se poursuit à l'IGM[46] avec Éric Laporte. Cette ressource est composée, pour le français, entre autres d'une soixantaine de tables[47] pour les verbes et, d'après [Sagot & Danlos, 2006][48], d'environ 30 tables pour les adjectifs.

Une autre ressource, principalement construite à partir du lexique-grammaire, est le Lexique des Formes Fléchies du Français[49] (Le*fff*), présenté dans [Sagot *et al.*, 2005]. Pour décrire notre étude sur l'effacement des compléments, nous utilisons cette ressource plutôt que le lexique-grammaire, car, selon notre point de vue, la structuration de ses informations est plus adéquate à notre approche. À ce sujet, les auteurs de [Sagot & Danlos, 2006] justifient l'intérêt des tables du Le*fff* en les comparant avec avec celles du lexique-grammaire, lesquelles « ne sont pas directement exploitables dans des systèmes d'analyse, une partie importante des informations n'étant compréhensible que moyennant de nombreuses connaissances implicites. À l'inverse, l'intérêt principal du Lefff est que sa structure et son format sont spécifiquement adaptés à une utilisation dans des systèmes de TAL. » Ces mêmes auteurs précisent toutefois que la qualité, la richesse et la couverture du Le*fff* n'atteignent pas celles des tables du lexique-grammaire. Notre approche sur cet aspect restant à l'heure actuelle uniquement théorique, nous ne sommes donc pas confrontés en pratique aux limites d'une telle ressource lexicale dans cette thèse.

Les adverbes et les pronoms ne disposent, quant à eux, pas d'information de sous-catégorisation dans le lexique-grammaire ou le Le*fff*. Leur fonctionnement semble être similaire à celui des adjectifs, ils pourront donc théoriquement être traités de la même façon, une fois qu'ils disposeront d'une ressource similaire.

Alors que de telles ressources voient classiquement une exploitation orientée vers un parsage syntaxique, [Bouiller *et al.*, 2005] pour le Le*fff*, notre approche se place en aval,

[45] Laboratoire d'Automatique Documentaire et Linguistique, `http://ladl.univ-mlv.fr`.

[46] Institut d'électronique et d'informatique Gaspard-Monge, `http://igm.univ-mlv.fr/`

[47] Chaque table correspond à un type de construction du verbe, selon le nombre, la nature et la position de ces compléments. Le nombre d'éléments par table est très variable, allant d'environ une dizaine pour à quelques centaines.

[48] Ces tables d'adjectifs ne sont pas encore accessibles librement comme le sont celles des verbes. Le site Internet de l'IGM n'y fait pas référence.

[49] Site Internet : `http://www.lefff.net/`

une fois que l'analyse a déjà été effectuée, et exploite sous un angle différent les traits du lexique.

3.3.1.2 La sous-catégorisation dans le Le*fff*

La sous-catégorisation, dans le Le*fff*, est représentée par une liste d'arguments, le sujet du prédicat inclus, dont les facultatifs sont mis entre parenthèse. Chaque flexion de chaque lemme dispose d'une entrée, c'est-à-dire une ligne, par type construction pour une tête donnée, dans le fichier de cette ressource. Un type de construction donné correspond généralement à une acception donnée du verbe.

Par exemple, pour le verbe *couper*, le lexique dispose, parmi l'ensemble des entrées du verbe, des entrées suivantes représentées dans le tableau 3.2.

#	FF	CT	Traits
...	...	...	...
1	couper	v	[...] 1 <suj:(sn),obj:(sn)> [...]
2	couper	v	[...] 1 <suj:(sn|sinf|scompl),obja:à-sn|à-scompl|à-sinf> [...]
...	...	...	...
3	coupera	v	[...] 1 <suj:sn,obj:(sn)> [...]
4	coupera	v	[...] 1 <suj:sn|sinf|scompl,obja:à-sn|à-scompl|à-sinf> [...]
...	...	...	...

TAB. 3.2 – Le verbe *couper* dans le Le*fff*.

Nous n'affichons pas toutes les informations présentes dans le lexique afin de nous concentrer sur celles qui nous sont utiles dans cette partie de notre travail. Dans l'exemple, nous avons extrait toutes les entrées du verbe *couper* à l'infinitif et à l'indicatif pour la troisième personne du singulier au futur. Pour chaque mode, temps et personne de ce verbe, au total deux constructions sont décrites.

Nous avons ajouté la première colonne afin de pouvoir faire référence, par un identifiant, à chaque entrée spécifiquement. Dans les colonnes, FF est la forme fléchie, c'est-à-dire celle qui est rencontrée dans le texte analysé, CT la catégorie lexicale, et $Traits$ l'ensemble des traits qui nous intéressent ici. La syntaxe contient un système de différentiation des verbes homonymes. Ainsi, le nombre[50] 1 placé avant les chevrons simples de toutes les entrées de l'exemple signifie qu'elles appartiennent toutes au même homonyme de couper. La syntaxe autorise aussi d'utiliser un nom mnémotechnique à la place du nombre. Puis, entre les chevrons est placée la liste des arguments : *suj* pour fonction sujet, *obj* pour fonction objet (direct) et *obja* pour fonction objet indirect introduit par à,

[50]Nous avons raccourci la notation du lexique qui précise le lemme du prédicat : *couper*_____1

ou fonction de type *à-objet*[51]. Après chaque fonction de complément sont précisées les catégories grammaticales réalisables (séparées par une barre verticale lorsque plusieurs sont possibles) : *sn* pour syntagme nominal, *sinf* pour syntagme infinitif, *scompl* pour syntagme phrastique fini (proposition complétive), ainsi que les mêmes réalisations précédées d'une préposition (*à* dans l'exemple).

Nous pouvons noter que les formes infinitives (1 et 2) voient leur sujet facultatif, car elles peuvent former un groupe infinitif nominal sans sujet réalisé : *Couper du beurre n'est pas difficile.* L'information qui nous intéresse concerne les autres arguments, et est identique pour les deux modes.

Nous disposons donc de deux constructions (entrées 1/3 et 2/4), lesquelles correspondent chacune à une acception.

Ces deux acceptions pourraient être associées aux acceptions 1 et 2 suivantes du TLFi :

1. acception 1 (entrées 1/3) : « Rompre un corps continu par l'intervention d'un instrument tranchant. »
2. acception 2 (entrées 2/4) : « Couper à qqc. (un désagrément, une tâche ennuyeuse, une punition, *etc.*). Y échapper. »

La première construction du verbe (entrées 1/3) admet un seul objet, facultatif car noté entre parenthèses, de type syntagme nominal. Cette construction peut donc se réaliser avec ou sans le complément d'objet comme dans les phrases *a* et *b* de l'exemple 3.10.

Exemple 3.10

a *Marie coupe le beurre.*

b *Marie coupe* $\otimes$ *.*

La seconde construction (entrées 2/4) nous permet de déduire que, si un complément d'objet indirect typiquement introduit par la préposition *à* est attaché à ce verbe, alors le verbe analysé est réalisé selon la seconde construction.

Nous considérons ici que l'analyseur syntaxique utilisé pour notre compression syntaxique exploite cette ressource en associant les mots de la phrase aux entrées correspondantes du lexique. Cette hypothèse implique que le lexique soit complet et juste pour les cas rencontrés. Pour les cas où il ne l'est pas, nous proposons une correction ou une complétion de l'information contenue dans le lexique. Nous verrons dans le prochain chapitre comment cette ressource peut être exploitée pour améliorer l'analyse des compléments de l'analyseur syntaxique que nous avons utilisé, et ainsi améliorer indirectement la qualité de nos compressions.

[51] Un tel complément est substituable par un syntagme prépositionnel de la forme à + pronom non-clitique

Maintenant que nous avons présenté les éléments structurels du Le*fff* qui nous intéressent, nous discutons des compléments obligatoires, puis de l'importance relative des facultatifs.

3.3.1.3 Les compléments obligatoires dans le Le*fff*

Un complément obligatoire, pour une position[52] et une construction données, est représenté par une liste de catégories lexicales sans parenthèse. Une fois l'entrée du lemme et de sa construction identifiée par l'analyse syntaxique, il suffit donc d'observer la présence ou non des parenthèses du complément considéré, dans sa position réalisée, pour savoir s'il est facultatif ou obligatoire.

Nous donnons ici quelques exemples, extraits du Le*fff*, corrigés pour certains, à cause de certaines imprécisions[53] du lexique, générés pour d'autres, du fait du manque de l'information de sous-catégorisation pour les adjectifs et adverbes.

Le tableau 3.3 présente trois cas de tête requérant un complément obligatoire, un pour chaque catégorie lexicale considérée.

#	FF	CT	SC
5	évoquer	v	obj:sn
6	enclin	adj	obja:à-sn|à-sinf
7	conformément	adv	obl:à-sn

Tab. 3.3 – Exemple de compléments obligatoires dans la syntaxe du Le*fff*.

Dans cet exemple, nous présentons uniquement les traits de sous-catégorisation (SC) et nous avons ôté l'information de l'argument sujet, non utile à notre approche. Nous avons dû corriger la fonction de l'objet de l'adjectif et renseigner celle de l'objet de l'adverbe. Pour cela, nous nous sommes appuyé sur la liste et les caractéristiques des fonctions proposées par les auteurs de [Sagot & Danlos, 2006]. Seules les entrées sous forme canonique[54] sont données car notre information de sous-catégorisation est identique pour toutes les autres formes fléchies.

Les entrées 5, 6 et 7 sont identifiables dans les phrases a, b et c de l'exemple 3.11. Supprimer les compléments sous-catégorisés produit une incohérence grammaticale, comme le montrent les phrases d, e et f du même exemple.

Exemple 3.11

[52]Pour les cas où il y a plusieurs compléments. Leur position est alors leur placement dans le syntagme verbal.

[53]Le lexique est en constante évolution. De telles imprécisions sont en voie d'être corrigées. La version 2.5 qui devrait être bientôt disponible devrait en corriger une partie.

[54]Forme canonique : infinitif pour les verbes, masculin singulier pour les noms et adjectifs.

a *Marie évoque une affaire.*

b *Marie est encline à la rêverie.*

c *Marie agit conformément à la loi.*

d * *Marie évoque* ⊗ *.*

e * *Marie est encline* ⊗ *.*

f * *Marie agit conformément* ⊗ *.*

Certains verbes admettent des homonymes, comme *voler*, *passer* ou *filer*. Il arrive que pour certains les différents homonymes partagent des constructions syntaxiques communes. Par exemple, pour *voler*, le tableau 3.4 présente les deux entrées du Le*fff*.

#	FF	CT	Traits
8	voler	v	[...] fly <suj:(sn)> [...]
9	voler	v	[...] steal <suj:(sn|sinf|scompl),obj:(sn),obja:(à-sn)> [...]

TAB. 3.4 – Les homonymes du verbe *voler* dans le Le*fff*.

Pour ce verbe, les identifiants des homonymes sont leur traduction en anglais. L'entrée 9 nous informe que les deux compléments du verbe sont facultatifs, et peuvent donc être supprimés sans nuire à la cohérence syntaxique. Cependant, les supprimer peut provoquer une grande perte d'information car la construction du syntagme verbal devient alors similaire à celle de l'entrée 8. Le sens du prédicat change donc radicalement, comme le montre l'exemple 3.12.

Exemple 3.12

a *Les oiseaux volent les graines.*

b *Les oiseaux volent* ⊗ *.*

Les phrases *a* et *b* de cet exemple n'expriment pas du tout la même action.

Ainsi, lorsque la suppression de compléments aboutit à une structure vérifiant une autre entrée du lexique, un changement considérable du sens du groupe verbal s'opère. Le complément en question voit alors son maintien indispensable. Cela est typiquement le cas pour deux homonymes, mais peut être rencontré pour deux acceptions d'un même verbe.

Ce comportement est généralisable aux autres catégories lexicales des têtes. Par exemple, dans le groupe adjectival *bon en maths*, la suppression du complément de l'adjectif *en maths* peut changer radicalement l'interprétation de ce constituant. Avant la suppression, le sens compris est celui d'une personne douée en mathématiques, alors qu'après la suppression, le sens compris peut être celui d'une personne généreuse.

Nous constatons que si l'analyse syntaxique est correcte et complète, notamment sur les informations de sous-catégorisation, alors la détermination des compléments obligatoires ainsi que leur préservation dans la compression de phrases ne pose théoriquement pas de problème.

Cependant, lorsque les informations sont partielles, plusieurs problèmes peuvent surgir et, selon les cas, des heuristiques peuvent être utilisées pour augmenter la robustesse du système. Dans certains cas, les informations requises peuvent même dépasser les objectifs de la ressource. Nous traitons ici des cas rencontrés lors de notre exploration du Le*fff*.

Informations de sous-catégorisation partielles. De nombreux verbes admettent plusieurs constructions dont certaines seulement disposent de compléments obligatoires (partir, aller, être...), ceci peut se produire, mais beaucoup plus rarement, pour les adjectifs. La ressource lexicale doit alors posséder toutes les constructions possibles du syntagme, afin de pouvoir faire face aux différents cas que l'on peut rencontrer dans les phrases.

Par exemple, le tableau 3.5 présente les entrées de l'adjectif *natif*, que devrait disposer le lexique.

#	FF	CT	SC	Exemple
10	natif	adj	loc:de-sn	Marie est native de Montpellier.
11	natif	adj		La patience est une qualité native de Marie.

Tab. 3.5 – Exemple de compléments obligatoires dans la syntaxe du Le*fff*.

L'abréviation *loc* signifie fonction locative. Les pronoms là, ici, là-bas sont substituables à ce type de complément.

Les entrées 10 et 11 peuvent être associées, respectivement, aux acceptions 1 et 2 suivantes du TLFi :

1. acception 1 : « Qui est originaire de tel endroit (lieu de résidence de la famille pendant un certain temps). »
2. acception 2 : « Que l'on possède en naissant. »

L'entrée 10 requiert obligatoirement un complément locatif gouverné par la préposition *de* alors que l'entrée 11 ne peut admettre aucun complément.

L'absence, dans le lexique, d'une entrée, comme la 10, mettrait en échec à la fois l'analyseur syntaxique, s'il exploite cette ressource pour déterminer l'attachement du complément à l'adjectif, mais aussi le compresseur syntaxique, qui utilise le lexique pour déterminer l'effaçabilité de ce complément. Si l'analyseur syntaxique attache tout de même le complément à l'adjectif, à défaut de lui avoir trouvé un autre attachement, le comportement prudent est alors de considérer le complément comme important, par défaut. Par

contre, si l'attachement est réalisé sur une autre tête, le compresseur syntaxique ne peut que se baser sur les informations de sous-catégorisation de cet autre attachement.

Une approximation, dans le lexique, qui spécifie un complément facultatif alors qu'il ne l'est pas, ne devrait pas mettre en échec l'analyseur syntaxique. En effet, ce dernier peut théoriquement toujours réaliser correctement l'attachement du complément, l'information de sous-catégorisation étant toujours présente. Cependant, cette approximation pose un problème au compresseur syntaxique qui, faisant naturellement confiance au lexique, envisage alors l'effacement du complément.

Enfin, un dernier cas problématique repéré est celui des interprétations au sens figuré du prédicat. À titre d'illustration, observons le couple de phrases de l'exemple 3.13.

Exemple 3.13

a Marie saute une barrière / une haie / un trou.

b Marie saute une classe / un échelon / une ligne.

Aucun des compléments n'est obligatoire. Cependant, dans la phrase où le sens du verbe est au figuré (*b*), la suppression du complément supprime par la même occasion la métaphore, modifiant ainsi profondément le sens de l'énoncé. Traiter de tels cas est beaucoup plus délicat que pour les précédentes difficultés, car une telle analyse requiert des informations beaucoup plus fines et spécifiques. Par exemple, pour le verbe sauter, l'analyseur devrait pouvoir estimer ce qui est *sautable* physiquement de ce qui ne l'est pas. L'étude de ces cas complexes ne fait pas l'objet de notre étude.

3.3.2 Les fonctions lexicales

Les fonctions lexicales (FL) constituent un trait linguistique mélangeant sémantique et syntaxe. Elles définissent des relations de proximité sémantique et de co-textualité entre lexies, ces relations ne pouvant être déterminées par des règles. Dans [Mel'čuk *et al.*, 1995], page 125, les auteurs justifient l'utilité des fonctions lexicales en expliquant notamment que, « de façon générale, les données sémantiques et les données syntaxiques, même prises ensemble, ne suffisent pas à déterminer entièrement l'utilisation d'une lexie vedette[55] ».

Ainsi une FL de proximité sémantique peut donner l'information que la *construction* est un type de *production*. Une FL de co-textualité peut donner l'information qu'une métaphore (figuratif) du nom *rideau* peut être *rideau de fumée*.

Les auteurs de [Mel'čuk *et al.*, 1995], page 127, définissent ainsi les FL :

[55] La lexie vedette fait référence à une entrée à part entière du lexique.

> [Les auteurs appellent] *fonction lexicale standard* [=FL] une fonction **f** qui associe à une lexie L un ensemble de lexies **f**(L) tel que les quatre conditions suivantes soient satisfaites :
>
> 1. Pour toute paire de lexies L_1 et L_2, les lexies $\mathbf{f}(L_1)$ et $\mathbf{f}(L_2)$ montrent des relations sémantico-syntaxiques (presque) identiques[56] à ces lexies :
>
> $$\frac{{}^{(}\mathbf{f}(L_1)^{)}}{{}^{(}L_1^{)}} \approx \frac{{}^{(}\mathbf{f}(L_2)^{)}}{{}^{(}L_2^{)}}$$
>
> 2. En règle générale, $\mathbf{f}(L_1)$ et $\mathbf{f}(L_2)$ sont différentes : $\mathbf{f}(L_1) \neq \mathbf{f}(L_2)$.
> 3. La fonction **f** a un nombre élevé d'arguments (= de mots-clés). [...]
> 4. La fonction **f** a un nombre élevé d'éléments dans sa valeur (= d'expressions).

Ce sont les relations de co-textualité qui nous intéressent ici, car beaucoup d'entre elles mettent souvent en relation une tête lexicale et un de ses modifieurs ou compléments. Soient T cette tête, G le constituant gouverné impliqué (modifieur ou complément) et C le constituant incluant directement T et G (défini par la projection maximale de T). Parmi ces relations de co-textualité, nous nous intéressons à celles où l'argument est un complément et détermine la réelle identité de C (figuratif, singulatif, collectif...). De telles relations sont presque toujours importantes, car supprimer G fait généralement perdre l'identité première de C.

À titre d'exemple, observons les quatre phrases de l'exemple 3.14.

Exemple 3.14

a On peut maintenant traiter le cœur du problème.

b On peut maintenant traiter le cœur ⊗ *.*

c On peut maintenant manger le cœur du poulet.

d On peut maintenant manger le cœur ⊗ *.*

Dans la phrase *a*, une FL « nom du centre » met en relation le nom *cœur* et le complément *du problème*. Dans la phrase *b*, le complément a été supprimé. On peut noter la perte considérable de contenu informationnel important. En effet l'interprétation du constituant *cœur du problème* est celle d'une notion abstraite, « la partie principale de », alors que l'interprétation du constituant *cœur* est ambiguë, hésitant entre la notion concrète, l'organe appelé cœur, et la notion abstraite. Par contre, dans la phrase *c*, où une telle FL

n'est pas présente, nous pouvons constater que supprimer le complément, en d, ne cause pas une aussi grande perte d'information.

Le tableau 3.6 regroupe les FL que nous avons jugées utiles à notre approche[57], selon les propriétés que je viens de présenter. Pour chaque ligne du tableau, sont spécifiés le numéro de la FL dans [Mel'čuk *et al.*, 1995], son nom, une courte description et quelques exemples. Le tableau regroupe uniquement des fonctions paradigmatiques.

Parmi les fonctions syntagmatiques, les FL verbales ($Oper_i$, Fun_i, $Labor_{ij}\ldots$) pourraient être exploités dans le cadre de notre approche. Cependant, nous ne les traitons pas ici, car celles qui pourraient nous intéresser peuvent généralement être traitées, en termes d'effacement, par les traits de sous-catégorisation que nous avons présenté en section précédente. Par exemple, certaines de ces FL peuvent être utilisées pour déterminer si un verbe est support. Le cas des verbes support est intéressant pour notre approche car l'information essentielle dans le groupe verbal n'est pas contenue dans le verbe lui-même mais généralement dans son complément direct, qui devient alors très important. Ainsi la FL $Oper_i$ met en relation les deux lexies suivantes ainsi : $Oper_1$(suprématie) = détenir, où la première pourra constituer la tête d'un complément direct de la seconde tête. Ce type de cas se traduit par un complément obligatoire pour le verbe *détenir* en termes de sous-catégorisation.

Pour les relations entre tête nominale et complément, il est possible de considérer un tel constituant « D_1 N_1 P (D_2) N_2 », comme un groupe nominal, sans complément, composé d'un déterminant complexe « D_1 N_1 P » et d'un nom « (D_2) N_2 ». Ainsi, la non-effaçabilité du déterminant complexe est justifiée par sa fonction de déterminant. Par exemple, pour le constituant *un rideau de fumée*, le déterminant complexe serait *un rideau de*.

Afin que ces FL soient exploitables dans un résumeur automatique, il est indispensable de disposer d'une ressource informatique disposant de ces informations relationnelles. Les auteurs de [Mel'čuk *et al.*, 1995] travaillent sur l'élaboration du DEC (Dictionnaire Explicatif et Combinatoire), qui est destiné à regrouper ce genre d'information à long terme. Ce travail a débouché sur une ressource lexicale numérique : le DiCo[58], ou dictionnaire de co-occurrences, qui est une base de données lexicales du français, développée depuis plusieurs années à l'Observatoire de linguistique Sens-Texte (OLST) de l'Université de Montréal par Igor Mel'čuk et Alain Polguère. La version du 1er août 2007 du DiCo contient 1075 lexies, c'est-à-dire des acceptions, ce qui représente une très petite partie de l'ensemble des acceptions du français. Le TLFi possède, par exemple, environ 300000 définitions. Cependant, renseigner correctement la grande quantité d'informations lexicales requises pour chaque entrée de DiCo, notamment pour les fonctions lexicales, est une tâche ex-

[57]Nous remercions sincèrement Didier Schwab, Docteur en Informatique de l'Université Montpellier 2, pour nous avoir orienté vers les FL à ce sujet.

[58]Accès au DiCo par son site Internet : `http://idefix.ling.umontreal.ca/dicouebe/`

#	Nom	Information supplémentaire	Exemples
(7)	Figuratif	nom métaphorique	rideau de fumée ; démon de la jalousie ; feu de la haine
(11)	Singulatif	unité minimale régulière de	grain de riz / de sel / de folie ; goutte de pluie / d'eau / de sang ; bouffée de fumée / de chaleur ; flocon de neige / d'avoine ; gousse d'ail
(12)	Collectif	ensemble régulier de	flotte de navires / de bateaux ; horde de barbares / de sauvages ; meute de chiens / de loups ; essaim d'abeilles / de sauterelles ; banc de poissons
(13)	Nom de chef	chef de	président de l'université ; directeur du théâtre
(14)	Nom d'équipe	équipe de	troupe de théâtre ; équipage d'avion
(15)	Nom de « démarrage »	germe/origine de	ferment / levain de la colère ; les premiers coups de feu de la guerre
(16)	Nom du centre	le centre de	le cœur du problème ; le plus profond de l'âme ; au cœur de l'hiver
(17)	Nom du pt. culminant	culmination de	comble de la joie ; paroxysme de la colère
(18)	DSAA	Dérivé sémantique adjectival actanciel	rempli, plein de mépris ; couvert de mépris

Tab. 3.6 – Les fonctions lexicales standard pertinentes à notre approche.

trêmement fastidieuse, ce qui explique sa faible couverture actuelle. Exploiter une telle ressource pour une tâche de résumé automatique est donc encore peu envisageable.

À noter que les FL ne peuvent se réaliser sur les *phrasèmes complets* (défini dans [Mel'čuk *et al.*, 1995]) car ils ne peuvent être décomposés en une lexie à laquelle on applique une fonction lexicale. Par exemple, le phrasème complet *le coup de pied de l'âne* qui signifie *la dernière attaque lâche contre un adversaire abattu*, ne correspond pas forcément à un certain type de coup de pied. Les semi-phrasèmes (comme *accepter / décliner une invitation*) peuvent quant à eux être analysés en termes de FL car ils peuvent être partiellement décrits en fonction de leurs constituants.

Nous traitons des phrasèmes complets dans la prochaine section.

3.3.3 Les autres traits linguistiques exploitables

Notre étude sur l'importance des constituants a révélé, en parallèle des fonctions syntaxiques, informations de sous-catégorisation et fonctions lexicales, un ensemble d'autres traits linguistiques exploitables pour réaliser une compression syntaxique des phrases. Dans cette section, nous présentons ces traits.

3.3.3.1 Les phrasèmes complets

Les phrasèmes complets sont aussi appelés expressions figées. Contrairement à une forme traditionnelle, le sens de la séquence d'un phrasème complet n'est pas le produit de celui de ses éléments composants. Son sens est associé à l'union de tous ses composants. Supprimer un des composants cause donc la perte de l'interprétation de son sens.

Prenons les phrasèmes complets *boire la tasse*, groupe verbal, et *pont aux ânes*[59], groupe nominal, et observons ce qui se passe lorsque nous supprimons leur complément dans les phrases de l'exemple 3.15.

Exemple 3.15

a *Marie a bu la tasse.*

b *Marie a bu* $\otimes$ *.*

c *Ce théorème est un pont aux ânes.*

d *Ce théorème est un pont* $\otimes$ *.*

Dans la phrase *a*, Marie a malencontreusement avalé une gorgée d'eau en se baignant, alors que dans la phrase *b*, elle a volontairement bu une boisson, très certainement alcoolisée. Dans la phrase *c*, le théorème est plus dur qu'il n'y paraît, alors que dans la phrase

[59] Cette expression signifie une difficulté apparente qui n'en est pas une.

d, on se demande bien ce que veut dire *être un pont*, pour un théorème. Les phrases *b* et *d* restent grammaticales, mais elles ont perdu une grande partie de leur sens.

Afin d'être exploitables dans un traitement automatique, les phrasèmes complets doivent constituer des entrées indépendantes dans un lexique.

Il faut bien noter qu'il ne s'agit pas ici de sous-catégorisation, car il n'est pas question de contrainte sur la catégorie lexicale. Pour l'expression *boire la tasse*, ce n'est pas le cas d'un verbe requérant un groupe nominal, mais plutôt d'un groupe verbal unitaire et indivisible.

3.3.3.2 L'article

« L'article défini s'emploie devant le nom qui désigne un être ou une autre chose connus du locuteur et de l'interlocuteur », [Grevisse, 1993–1997]. Ainsi, lorsqu'un tel déterminant est utilisé, le locuteur doit s'assurer que l'interlocuteur identifie bien cette entité dont il parle. S'il y a une ambiguïté sur l'entité, le locuteur peut utiliser un modifieur de nom pour désambiguïser. Si le modifieur du nom est supprimé, alors, en toute logique, l'ambiguïté devrait réapparaître. Le modifieur de nom semble donc important lorsque le nom est précédé d'un article défini.

L'exemple 3.16 illustre cette utilisation de modifieur.

Exemple 3.16

a Le gros chien a mangé votre chat.

b Le ⊗ chien a mangé votre chat.

Dans la phrase *a*, l'interlocuteur est renseigné sur un chien particulier qui avait mangé son chat. Dans la phrase *b*, l'interlocuteur ne sait pas quel chien a mangé son chat.

Ce principe de désambiguïsation d'entité peut être retrouvé pour l'article indéfini, dans le cas peu fréquent des généralités. L'exemple 3.17 illustre ce genre de cas.

Exemple 3.17

a Un gros chat est un chat qui mange beaucoup.

b Un ⊗ chat est un chat qui mange beaucoup.

La suppression du modifieur du nom cause ici aussi une perte importante de contenu informationnel.

L'article défini est facilement détectable automatiquement grâce à sa morphologie. Cet indice d'importance est donc assez facilement exploitable. Pour le cas des généralités avec un article indéfini, cette analyse plus complexe ne fait pas l'étude de notre travail.

3.3.3.3 Les éléments incidents

[Grevisse, 1993–1997] définit un élément incident comme « une espèce de parenthèse par laquelle celui qui parle ou écrit interrompt la phrase pour une intervention personnelle. » Son caractère facultatif sur le plan syntaxique comme le plan sémantique est naturellement intéressant dans le cadre d'une compression de phrases, comme nous pouvons le constater dans les trois phrases de l'exemple 3.18.

Exemple 3.18

a *Soit dit entre nous, il n'est guère consciencieux dans son travail.*

b *Or l'ancien ministre savait que les janjawids "attaquaient les populations civiles et commettaient des crimes", écrit le procureur.*

c *Il eût été, probablement, très fort de demander sa main.*

d *Je lui dis, façon de plaisanter, que je ne voulais plus le voir.*

Dans la phrase *b*, l'élément incident est une proposition incise, fréquente dans les articles journalistiques.

Le cas de l'élément incident est toutefois en marge de notre approche, car il n'a généralement pas de fonction syntaxique dans la phrase. Il ne peut donc être un modifieur ou un complément.

De plus il semble être difficile de le détecter automatiquement. Dans un analyseur syntaxique, si aucun traitement particulier ne lui est consacré, l'élément incident risque fort de se voir attribuer une fonction de modifieur. Il sera alors considéré comme un modifieur détaché, ce qui constitue en fait une bonne approximation dans la tâche de compression de phrases, car, comme nous allons le voir maintenant, le modifieur détaché est un bon candidat à la suppression.

3.3.3.4 Le modifieur du nom détaché

« Quand l'épithète (adjectif et surtout participe) ne restreint pas l'extension du nom, mais apporte une indication complémentaire, descriptive ou explicative, elle est souvent séparée de ce nom », [Grevisse, 1993–1997]. Cette indication complémentaire peut être typiquement ôtée pour la production d'un résumé. Tout comme l'épithète, tout modifieur du nom, incluant notamment les appositions nominales, peut être détaché du nom, comme nous pouvons le voir dans les phrases de l'exemple 3.19.

Exemple 3.19

a *Félix, vif et svelte, chasse toute la journée.*

b *Félix, maintenant fatigué, se repose sur le canapé.*

c Félix, qui digère son repas, n'a pas bougé depuis deux heures.

d Félix, grand chat brun, fait la fierté de Marie.

e Félix, le grand chat brun de Marie, n'a pas de rival dans son quartier.

Les phrases *d* et *e* illustrent le cas de l'apposition nominale, la première avec un groupe nominal sans déterminant, et la seconde avec.

Le détachement peut aussi s'appliquer à certains modifieurs de la proposition, cependant, il semble influer moins sur l'importance de l'élément détaché.

Le détachement est séparé par une (ou deux) virgules du constituant auquel il s'attache, ce qui le rend assez facilement localisable automatiquement dans la phrase. Un traitement particulier pour le modifieur détaché est donc intéressant dans la tâche de compression de phrases.

3.3.3.5 La position des constituants dans la phrase

Selon où sont placés certains constituants gouvernés, leur importance peut varier.

« Placer en tête de phrase, en position de thème, le circonstant de phrase[60], est un procédé fréquent dans les textes narratifs (il s'agit alors de circonstants de temps) et descriptifs (il s'agit des circonstants de lieu). En position de thème, ces circonstants de phrase constituent le cadre qui organise la cohérence du texte », écrit Roberte Tomassone dans [Tomassone, 2001]. Une importance accrue est donc attribuée à ce type de circonstants. La position du constituant étant naturellement facile à obtenir, cette information pourra facilement être exploitable.

3.3.3.6 La négation et l'interrogation

Certains éléments de la phrase permettent d'appliquer une négation sur d'autres éléments. Parmi l'ensemble des mots de la langue, ce sont principalement certains modifieurs adverbiaux (*ne*, *pas*, *jamais*, *plus*...) qui marquent la négation. Cependant, d'autres catégories lexicales peuvent la marquer ([Grevisse, 1993–1997], page 1475), comme :

- la préposition *sans* ;
- la locution conjonctive de subordination *sans que* ;
- la conjonction de coordination *ni* ;
- les mots-phrases *non* et *nenni*

La négation peut être un modifieur de plusieurs catégories différentes de tête lexicale, par exemple :

- verbe : Marie ne vient pas.

[60]Nous appelons circonstants de phrase nos modifieurs de la proposition, lesquels précisent, pour les verbes d'action, les circonstances de l'action.

– adjectif : Les chats de Marie, jamais fatigués, s'amusent dans le jardin.
– préposition : Non sans peine, Marie a puni son chat.
– pronom : Son avis, non celui de Marie, doit prévaloir.

Ôter une négation change profondément le sens du constituant affecté, lui attribuant une signification opposée. Il est donc important de la détecter pour ne pas la supprimer dans une compression de phrases.

Au sujet des modifieurs de la proposition, dans [Tomassone, 2001], l'auteur écrit que *sont considérés comme effaçables, les circonstants qui n'affectent ni la négation, ni l'interrogation.* Ainsi ce ne sont pas uniquement les éléments portant la négation qui sont touchés par la modification d'importance de la négation, mais aussi certains éléments affectés par cette dernière.

Nous observons ici que la forme interrogative influe aussi sur l'importance de certains constituants. Par exemple lorsque la question porte sur le modifieur ou le complément comme dans les phrases de l'exemple 3.20.

Exemple 3.20

– *Pour le modifieur de la proposition : Travaille-t-il à Paris ?*
– *Pour le modifieur du verbe : Mange-t-il beaucoup ?*
– *Pour le complément de l'adjectif : D'où Marie est-elle native ?*

La forme interrogative ainsi que la négation sont facilement détectables automatiquement à l'heure actuelle. Ces informations sont donc raisonnablement exploitables.

3.4 Esquisse d'un modèle computationnel compatible avec le résumé par compression syntaxique de phrases

Les trois premières sections de ce chapitre ont décrit les différents éléments de notre méthode de compression syntaxique des phrases. Une des caractéristiques souhaitées de cette méthode est de pouvoir être mise en œuvre dans un système informatique, dans le but d'être automatisée. Nous l'avons ainsi orientée en exploitant des informations dont l'extraction automatique est à la portée des techniques actuelles du domaine. Cette méthode se veut donc être un modèle computationnel de résumé par compression syntaxique.

Dans cette section nous décrivons ce modèle, qui réunit les éléments théoriques des trois premières sections. Nous présentons tout d'abord les objets manipulés, section 3.4.1, puis nous proposons un algorithme de compression syntaxique de phrases exploitant ces objets, section 3.4.2.

3.4.1 Nos objets linguistiques

La granularité des segments textuels que nous avons choisis pour la tâche de résumé est la phrase. Puis, lorsque nous avons considéré le niveau intra-phrastique, nous avons opté pour le constituant comme unité textuelle d'analyse pour la compression syntaxique, section 3.1.1. Pour déterminer quels constituants sont effaçables (syntaxiquement), nous avons choisi d'exploiter l'information de fonction syntaxique de ces constituants. L'étude qui en a suivi, section 3.2, nous a permis de définir deux fonctions syntaxiques, chacune admettant des propriétés d'effacement différentes : le complément et le modifieur. Ces deux fonctions définissent donc les objets de notre modèle. Ces deux termes sont régulièrement employés dans la littérature du domaine, mais leur définition peut varier assez profondément d'une théorie à une autre. Nous synthétisons ici les principales caractéristiques de ces deux objets que nous leur avons attribuées.

Le complément. Cette fonction définit un constituant gouverné, sous-catégorisé par sa tête. Nous avons présenté des compléments pour chacune des têtes syntaxique ou fonctionnelle introduite dans ce chapitre, c'est-à-dire le nom, le verbe, l'adjectif, la préposition, l'adverbe, le pronom et la proposition (la tête fonctionnelle est en fait I^0, mais nous utilisons la proposition comme représentant de cette tête pour des raisons de clarté). Pour chaque tête, nous avons défini dans quelles conditions leurs compléments pouvaient être effacés. En voici un résumé tête par tête :

- le nom : le déterminant, que nous avons classé parmi les compléments du nom, est le seul qui ne peut être effacé ;
- le verbe, l'adjectif et l'adverbe : la possibilité d'effacement du ou des compléments dépend de la construction syntaxique de l'acception du lemme considéré ;
- la préposition : le complément n'est jamais effaçable ;
- le pronom : le complément n'est effaçable que pour certains types de pronoms ;
- la proposition : nous considérons les groupes sujet et prédicats comme les seuls compléments possibles de la tête, lesquels ne peuvent jamais être effacés.

Enfin, pour certains compléments effaçables, nous avons repéré des critères linguistiques d'importance de ces compléments. Lorsqu'une fonction lexicale de co-textualité entre un nom et son complément est présente (section 3.3.2), ce dernier voit son importance généralement accrue. Si un complément est impliqué dans un phrasème complet (section 3.3.3.1), alors il devient indispensable à la bonne compréhension du constituant, son importance est donc très élevée.

Le modifieur. Cette fonction définit un constituant gouverné, non sous-catégorisé par sa tête, systématiquement effaçable. Il correspond à l'union des spécifieurs et adjoints de

la théorie GB, aux exceptions des déterminants, que nous classons parmi les compléments, et des adjoints du verbe, que nous classons parmi les modifieurs de la proposition. Tout comme pour le complément, nous avons présenté des modifieurs pour chaque tête introduite dans ce chapitre puis repéré des critères linguistiques d'importance de ces modifieurs. Voici ces critères et leur impact :

- l'information de sous-catégorisation, lorsqu'elle est partielle, peut conduire à des ambiguïtés d'attachement, qui dans certains cas peuvent être décelées par le compresseur syntaxique, lequel peut alors prévenir une compression trop dégradante (section 3.3.1.3) ;
- lorsque le modifieur est inclus dans un phrasème complet, alors, tout comme pour le complément, le modifieur devient indispensable ;
- le modifieur du nom, s'il est détaché, constitue généralement un bon candidat à la suppression (section 3.3.3.4) ;
- les modifieurs de la proposition de type circonstants de temps ou de lieu, lorsqu'ils sont placés en tête de la phrase, dans un texte de genre narratif, sont généralement importants (section 3.3.3.5) ;
- dans un groupe nominal, si le déterminant est un article défini, alors le modifieur du nom est généralement plus important (section 3.3.3.2) ;
- les modifieurs marquant la négation ne peuvent être effacés sans nuire profondément à la compréhension (section 3.3.3.6) ;
- les modifieurs et compléments affectés par la négation ou l'interrogation doivent aussi être conservés (section 3.3.3.6).

3.4.2 Notre algorithme de compression syntaxique

Nous proposons maintenant un algorithme de compression syntaxique, exploitant nos objets et leurs propriétés. Pour une phrase donnée, le résultat de cet algorithme dépend principalement de deux facteurs :

1. le niveau de conservation de contenu informationnel important souhaité ;
2. la qualité des informations linguistiques extractibles de la phrase.

Le premier facteur est déterminé par le but applicatif du résumé, selon le taux de compression et la qualité de la compression produite souhaités. Le second facteur est déterminé par le niveau d'avancée des analyseurs syntaxiques, ainsi que par la qualité des ressources linguistiques exploitables disponibles.

Nous avons choisi l'arbre syntagmatique vérifiant notre système de règles structurelles (3.5–3.9), défini en section 3.2.2.7, comme base pour la donnée traitée dans notre algorithme. L'arbre actuel vérifiant notre système de règles, porte alors en sa structure

les informations nécessaires pour localiser nos objets, les compléments et modifieurs. Les compléments sont les frères de la tête (X^0), et les modifieurs les frères des X'.

Cependant, afin de faciliter le traitement de l'arbre, nous préférons le transformer en arbre syntagmatique, c'est-à-dire où chaque nœud interne est un constituant. Pour cela, il suffit de fusionner tous les X' avec leur XP (leur premier parent XP). La figure 3.8 illustre cette fusion au sein d'un constituant.

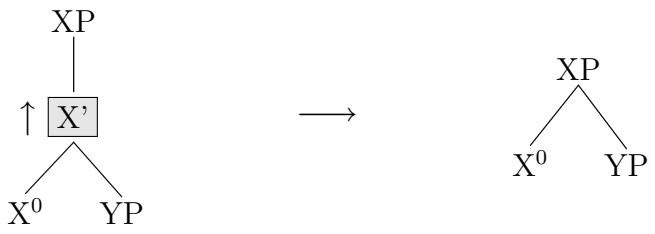

FIG. 3.8 – Fusion d'un X' avec un XP.

Afin de ne pas perdre l'information des compléments et modifieurs, nous attachons à chacun de ces nœuds l'information de la fonction, sous la forme d'un couple variable – valeur : $FS = C$ ou $FS = M$, avec FS pour fonction syntaxique, C pour complément et M pour modifieur. Nous supprimons également la tête fonctionnelle I^0 qui n'est plus nécessaire à ce stade, sachant qu'elle n'a pas de réalisation lexicale et ne porte pas d'information utile au traitement, elle ne pourra donc pas influencer le résultat de l'algorithme.

L'arbre de la figure 3.9 illustre cette transformation, à partir du constituant *sous une pluie diluvienne*, issu de l'arbre de la figure 3.7.

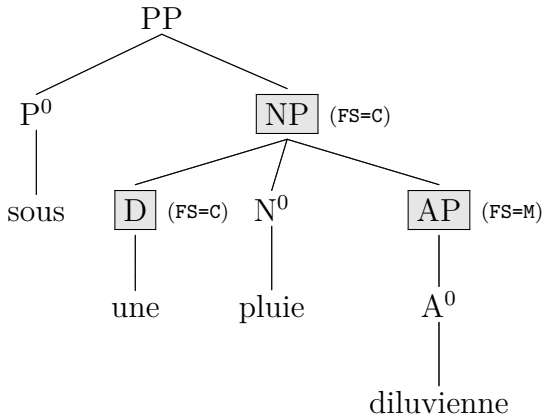

FIG. 3.9 – Exemple de fusion des X' avec les XP.

La catégorie lexicale de la tête modifiée par un complément ou un modifieur est accessible directement sur l'étiquette du nœud de leur père. Ainsi, le constituant *une pluie*

diluvienne, NP, a pour père un groupe prépositionnel (PP) et pour fonction complément ($FS = C$), il est donc un complément de la proposition.

Les autres informations linguistiques utiles à notre approche (sous-catégorisation, FL, autres traits) sont, tout comme la fonction, associées à chaque nœud XP, c'est-à-dire à chaque constituant, par un couple variable – valeur.

Chaque feuille de nos arbres est un mot de la phrase. Chaque arbre correspond à la structure profonde de la phrase, car il vérifie les règles structurelles de notre système. Dans certaines constructions particulières, certaines modifications de la structure peuvent intervenir, faisant alors diverger la structure profonde de la structure de surface. Cela implique que l'ordre des feuilles peut ne pas respecter l'ordre des mots de la phrase. Or, il est important de conserver la place relative de chaque mot dans la phrase afin de pouvoir le restituer au bon endroit. Pour cela nous ajoutons une variable d'indice du mot à chaque feuille. L'arbre de la figure 3.10, basée sur la phrase *Viens-tu demain ?*, illustre une telle variable d'indice, appelée Ind, pour le cas de la forme interrogative.

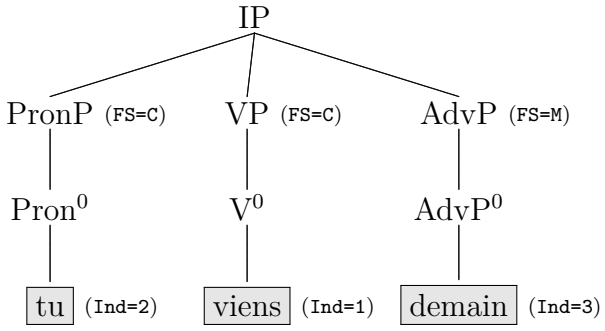

FIG. 3.10 – Exemple d'indice de position des mots dans la phrase.

Enfin, nous appelons *arbre enrichi*, notre arbre syntagmatique obtenu après les transformations et ajouts d'information que nous venons de décrire.

La méthode de compression consiste alors à parcourir les nœuds de l'arbre en profondeur, puis pour chaque complément et modifieur qui vérifie les propriétés d'effacement désirées, élaguer sa branche de l'arbre (ne pas la parcourir), enfin, une fois l'arbre entièrement parcouru, recomposer la phrase à partir des feuilles restantes, c'est-à-dire les mots de la phrase, en respectant leur ordre d'origine.

L'algorithme 1 est une définition semi-formelle de cette méthode de compression.

La procédure *ÉlagueArbre(n)* réalise l'élagage de l'arbre, en testant, pour chaque complément et modifieur, s'il vérifie nos conditions de suppression. Comme nous avons vu en début de cette section, ces conditions sont dépendantes de deux facteurs.

Le premier, le niveau de conservation de contenu informationnel important souhaité, implique la conservation des constituants non effaçables et jugés importants, et la suppression de ceux jugés effaçables et non importants. Quant aux constituants pour lesquels

Données : P = phrase à compresser
Résultat : P' = phrase compressée

// Analyse syntaxique qui retourne l'arbre enrichi;
$A \longleftarrow$ AnalyseSyntaxiqueEnrichie(P);

// Élagage de l'arbre, selon nos critères de compression;
ÉlagueArbre(racine de A);

$LF \longleftarrow$ linéarisation de l'arbre A par production de la liste des feuilles dans un parcours en profondeur;

$P' \longleftarrow$ tri des éléments de la liste LF par ordre d'indices croissant;

Algorithme 1 : CompressionSyntaxique.

Données : n = un nœud de l'arbre à élaguer

pour *chaque fils f de n* **faire**
  si *f est une feuille* **alors**
    retour;
  sinon
    si *($FS(f) = C$* **ou** *$FS(f) = M$)* **et** *f vérifie nos conditions de suppression* **alors**
      supprimer f et n;
    sinon
      ÉlagueArbre(f);
    fin
  fin
fin

Procédure `ÉlagueArbre(`n`)`.

nous ne disposons pas de critère d'importance, ils seront supprimés si la conservation du contenu important est moins importante que le taux de conservation souhaité.

Le second facteur, la qualité des informations linguistiques extractibles, influence le premier facteur, en proposant soit moins de critères d'effaçabilité ou d'importance des constituants, soit des critères partiellement ou entièrement erronés.

La gestion pratique de ces deux facteurs dépend de la mise en œuvre du système de compression. Nous présenterons le fonctionnement de notre procédure d'élagage, utilisée dans notre système de compression, dans le prochain chapitre.

3.5 L'influence du genre de texte sur l'importance des modifieurs et compléments

Lors de nos premiers tests de compression manuelle (notamment dans notre étude préliminaire décrite en introduction), nous avons remarqué que le genre du texte semblait avoir une influence sur l'importance des différents modifieurs et compléments dans la phrase. Cela rejoint les propos de François Rastier dans [Rastier, 2002], comme vu au chapitre précédent, section 2.2.1.5.

Suite à cette observation, nous avons réalisé des tests de suppression de certains constituants, à partir de textes de genres variés, en nous basant sur leur fonction syntaxique, et en estimant qualitativement les pertes de cohérence discursive et de contenu important dans les phrases comprimées. Ces premiers tests étant essentiellement pour se faire une idée, le protocole de constitution des corpus n'était pas particulièrement important. En substance nous avons considéré :

- le corpus de dépêches journalistiques, utilisé dans [Chauché *et al.*, 2003] pour la catégorisation (plus d'un million de phrases, de style hétérogène et journalistique) ;
- quelques articles scientifiques en biologie ;
- quelques textes narratifs (romans, contes, ...).

Nous avons fait des *sondages* en prenant au hasard des textes ou des sous-textes, et nous avons essayé la méthode d'effacement, à la main, avant même d'avoir à l'évaluer automatiquement, pour savoir si la méthode était capable de résister à une première *plongée* dans les textes. Les constatations que nous avons faites, à partir des données considérées sont les suivantes.

Dans les textes du genre article scientifique ou énoncé technique, chaque constituant se révèle avoir beaucoup plus d'importance que dans un texte narratif. Prenons par exemple le terme *hormone de synthèse* (article de biologie), il serait très ennuyeux de supprimer le complément de nom. De la même manière, il serait gênant d'amputer la phrase *Un vent de 50 kmh soufflera sur le Golfe du Lion.* de son complément circonstanciel de lieu

(*le Golfe du Lion*), dans les dépêches météo. En revanche, dans *L'étalon noir broutait, tranquillement, en remuant la queue, près de l'enclos principal.*, il est tout à fait possible de réduire cette phrase sans perte d'information risquant d'en transformer le sens. La raison est que les auteurs de textes narratifs ajoutent de nombreuses informations à caractère essentiellement descriptif qui aident le lecteur à être transporté dans l'histoire mais qui ne sont pas indispensables à la compréhension du cœur de cette dernière. Alors que dans un article scientifique ou technique, beaucoup de constituants ont un rôle important à jouer dans la compréhension du discours.

Cette constatation nous a incité explorer l'impact du genre sur l'importance des constituants dans nos expérimentations décrites au chapitre 5.

3.6 Les limites de la localisation du contenu important

Notre approche explore des traits linguistiques au sein de la phrase dans le but de déterminer l'importance de ses constituants. Cela nous a permis d'établir des classes de constituants effaçables puis de proposer des critères de détermination de l'importance de ces constituants effaçables.

Cependant, lorsqu'il s'agit de réaliser la compression de manière automatique, les critères que nous avons proposés ne sont pas toujours facilement calculables, car les informations dont ils dépendent ne sont pas toujours facilement extractibles de la phrase. Par exemple, nous discutons en section 3.3.1 de l'exploitation d'un lexique tel le Le*fff*, cependant ce lexique est encore jeune et incomplet sur de nombreux points. Les informations de sous-catégorisation des verbes sont partielles et approximatives, celles des adjectifs presque nulles et celles des adverbes inexistantes. De plus, faire correspondre par un traitement automatique les informations du lexique aux éléments de la phrase n'est que très partiellement réalisable.

Enfin, d'autres critères seraient exploitables, lesquels pourraient être basés sur les sources d'information suivantes :

1. d'autres informations au sein de la phrase ;
2. des informations contextuelles ;
3. les préférences du récepteur du résumé.

Pour le premier type de source, ce peut être des informations dont l'extraction relève de champs de recherches encore ouverts à l'heure actuelle (touchant à la pragmatique par exemple), mais aussi des données simplement non traitées dans ce travail, lequel ayant dû se fixer des limites raisonnables dans le cadre d'une thèse. Pour le second type de

source, de nombreuses approches exploitent ce genre d'information (thème du document, chaînes lexicales, fréquence des termes...), parcourues dans le chapitre précédent. Comme nous l'avons vu dans nos objectifs en introduction de ce chapitre, ces techniques sortent du cadre de notre approche. Enfin, pour le troisième type de source, l'intérêt est de déterminer, au moins partiellement, le choix des constituants importants selon l'avis de la personne à qui est destiné le résumé.

Notre approche, tout en permettant de réaliser un système de compression automatique selon certains critères, voit par ailleurs des limites naturelles mais inévitables, dans le contexte actuel. Cette constatation nous a incité à explorer une piste parallèle au résumé automatique : le résumé semi-automatique.

Nous nous sommes alors orientés vers la production d'un système de compression syntaxique à deux facettes : un compresseur purement automatique et un autre faisant intervenir les choix d'un utilisateur à travers une interaction. L'intervention de l'utilisateur a pour but de corriger les erreurs du système sur la détermination de l'importance des constituants.

Ces deux facettes de notre résumeur divergent sur un autre point important : le but applicatif. Les deux facteurs responsables de cette divergence sont le temps de compression, beaucoup plus court sans intervention de l'utilisateur, et la qualité de compression, nettement meilleure avec intervention de l'utilisateur. La version automatique s'oriente alors vers des applications demandant une production de compressions en grande quantité et dans un petit délai de temps, défavorisant alors la qualité des résumés, alors que la version semi-automatique prend plutôt la direction d'applications favorisant la qualité de la compression produite, au détriment du temps de production nécessaire pour chaque compression.

Nous décrivons les deux modes de fonctionnement de notre résumeur dans le prochain chapitre.

3.7 Conclusion

Ce chapitre constitue l'étude théorique des bases linguistiques de notre approche de compression syntaxique de phrase. Nous avons commencé par définir la compression syntaxique en constituants ainsi que nos objectifs vis-à-vis de ce type de résumé.

La première partie a eu pour but de définir les constituants supprimables sur le plan syntaxique, selon leur fonction syntaxique. Nous avons ainsi abouti à une classification des constituants gouvernés : les modifieurs et les compléments. Les premiers sont toujours effaçables syntaxiquement, alors que l'effaçabilité des seconds dépend de leur tête syntaxique. Ce premier critère de compression permet de réaliser un premier palier de

compression, nécessitant peu d'informations syntaxiques, ce qui le rend assez facilement programmable, mais produisant des compressions dont la cohérence syntaxique n'est pas toujours garantie, et donc la cohérence sémantique n'est pas directement considérée.

Dans la seconde partie, nous avons étudié un ensemble de traits linguistiques que nous avons jugés exploitables pour aider à déterminer l'importance des modifieurs et compléments. La sous-catégorisation nous a permis de déterminer quels compléments étaient indispensables à la cohérence syntaxique de la phrase. Les autres traits nous ont donné des critères d'importance du contenu informationnel de différents modifieurs et compléments.

Nous avons ensuite, dans la troisième partie, décrit et résumé notre approche dans un modèle computationnel, qui sera utilisé comme base de développement de notre prototype, lequel est décrit dans le prochain chapitre.

Nos tests de compression manuelle nous ont révélé l'impact du genre textuel sur l'importance des différents modifieurs et compléments dans la phrase. Nous avons alors ouvert la discussion sur cette influence du genre et nous engageons à l'explorer d'avantage par une expérimentation avec notre système de compression sur un corpus mélangeant plusieurs genres.

Enfin nous avons ouvert une piste à explorer de manière parallèle, qui a pour but d'améliorer la détermination de l'importance des constituants, par une interaction entre un utilisateur et notre système de compression automatique.

Dans le prochain chapitre nous décrivons le fonctionnement de notre système de compression ainsi que l'outil de compression que nous avons développé.

4

Conception du compresseur de phrases

Sommaire

4.1 Introduction 87
4.2 Architecture 88
4.3 Analyse syntaxique 91
4.4 Le compresseur de phrases COLIN 110
4.5 Conclusion 128

4.1 Introduction

Évaluer l'efficacité d'une théorie sur un système automatique nécessite le développement d'un outil informatique, puis la réalisation d'expérimentations. Dans les débuts de nos travaux, nous avons dû réaliser nos premiers tests de compression manuellement. Ce travail fut extrêmement coûteux en temps et en efforts cognitifs, car il nous imposa d'extraire toutes les informations nécessaires à la compression, puis de les traiter rigoureusement, selon la méthode définie. Dans le cas de notre approche, les informations relèvent des fonctions syntaxiques des constituants, ainsi que de nombreuses informations linguistiques, vues au chapitre précédent. Le traitement consiste quant à lui à appliquer la méthode de sélection des constituants susceptibles d'être supprimés, au seuil d'importance souhaité, tout en incluant toutes les inclusions possibles de constituants, puis de supprimer ces constituants.

Cette décomposition du traitement en deux étapes nous orienta vers une technique manuelle. L'approche qui nous vint naturellement fut d'imprimer les documents de test[61],

[61] Ces documents sont typiquement ceux utilisés lors de notre étude préliminaire décrite en introduction

puis de surligner les constituants potentiellement supprimables. Afin de différencier les différents types de constituants, nous avons défini un code de couleur, avec une table d'association couleur – type de constituant. Représenter correctement l'inclusion des constituants, puis appliquer les filtres de suppression selon les seuils désirés nous incita rapidement à développer un outil automatique pour réaliser cette tâche.

Ce chapitre s'organise ainsi autour de la conception d'un compresseur syntaxique de phrases, en voici l'organisation.

Nous commençons par présenter, section 4.2, l'architecture globale de notre système de compression syntaxique de phrases, qui définit les étapes principales de la méthode de compression ainsi que les outils et ressources exploitées. Ce système se veut autant que possible indépendant des différents choix des outils et de la mise en œuvre. Le reste du chapitre décrit nos choix à ce sujet, lesquels restent toutefois limités par les ressources linguistiques existantes et exploitables.

Nous continuons en détaillant, section 4.3, SYGFRAN, l'analyseur syntaxique utilisé dans notre système. Cet analyseur utilise un ensemble de règles créées manuellement, appliquées par un système opérationnel de transformation de structures arborescentes. Nous présentons alors les caractéristiques globales de SYGFRAN, qui ont motivé notre choix pour cet analyseur.

Enfin, section 4.4, nous présentons COLIN (COmpresseur LINguistique), notre outil informatique de compression de phrases, exploitant le résultat de l'analyseur syntaxique.

4.2 Architecture

Dans cette section est décrite l'architecture globale d'un système de compression syntaxique de phrases, selon les éléments théoriques développés dans le chapitre précédent. Le schéma de cette architecture est représenté dans la figure 4.1.

Le processus complet s'organise en trois principales étapes que nous décrivons maintenant.

4.2.1 Première étape : analyse syntaxique

La donnée source est le texte à compresser. Ce texte est récupéré en entrée d'un analyseur syntaxique, qui va réaliser un découpage en constituants de chaque phrase, puis un marquage de ces constituants selon les informations utiles à notre compression. Pour cela, l'analyseur doit disposer de *ressources linguistiques* adéquates (définies dans le précédent chapitre, structurées dans un lexique) : les traits de sous-catégorisation (SC),

du chapitre précédent.

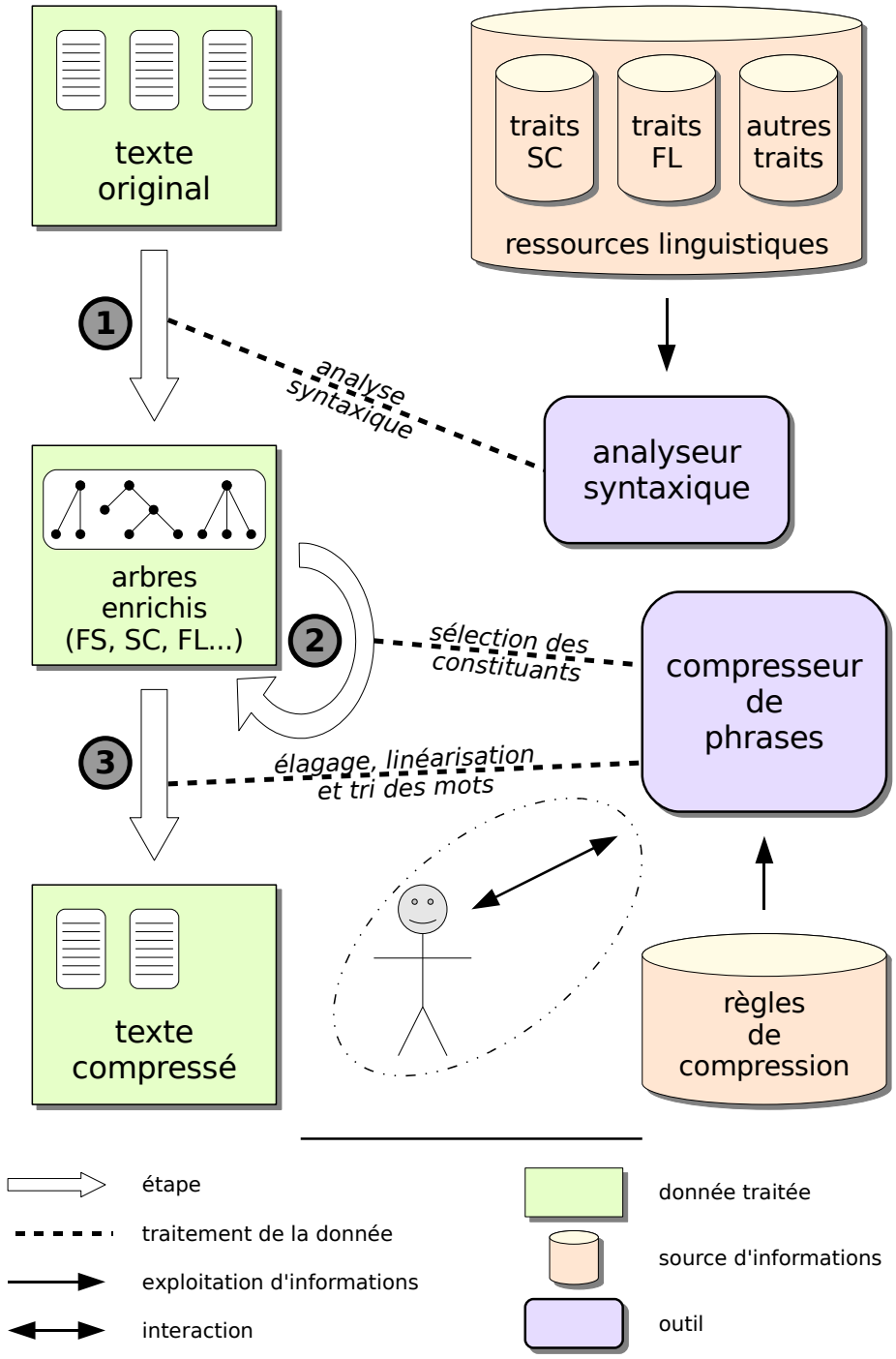

FIG. 4.1 – Architecture d'un système de compression syntaxique de phrase selon notre méthode de compression théorique.

fonctions lexicales (FL) ainsi que les autres traits. La tâche de l'analyseur est, pour chaque phrase :

- de déterminer la délimitation de chaque constituant ainsi que leur inter-inclusion ;
- de déterminer la fonction syntaxique (FS) des constituants gouvernés visés, c'est-à-dire les compléments et modifieurs (selon notre définition) ;
- d'affecter aux compléments et modifieurs les informations extraites des ressources (compléments obligatoires, FL, phrasèmes complets. . .)

tout cela dans une arborescence d'arbres syntagmatiques enrichis. Cette structure est alors la donnée en sortie de l'analyseur.

4.2.2 Seconde étape : sélection des constituants

Le compresseur syntaxique récupère alors cette structure. Pour effectuer la compression, il doit disposer de notre méthode de compression, ici structurée sous forme de règles. Ces dernières décrivent le mode de sélection des constituants susceptibles d'être effacés, selon les informations linguistiques attribuées à l'étape précédente. Plusieurs sélections différentes peuvent être réalisées, selon le taux de compression et/ou le niveau de conservation du contenu important souhaités. Les règles de compression sont donc paramétrables.

Par exemple, si le taux de compression prime, seront supprimés tous les compléments et modifieurs pour lesquels nous ne disposons pas d'indice d'importance.

Le système peut éventuellement intégrer une interaction avec un utilisateur, qui pourra intervenir dans les choix de sélection des constituants. Cela peut s'effectuer soit de manière globale, en intervenant sur chaque paramètre des règles, soit de manière locale, en validant ou invalidant la sélection de chaque constituant.

La possibilité de l'interaction nous a poussé à découper l'étape d'élagage en deux phases :

- la sélection des constituants (la deuxième étape du schéma), jusqu'à satisfaction ;
- l'élagage proprement dit de l'arbre (la troisième étape).

Dans le cas d'un résumé purement automatique, les deux phases se suivent sans interruption.

4.2.3 Troisième étape : compression de phrases

À partir de la structure de donnée intermédiaire de la précédente étape, les constituants qui ont été sélectionnés sont enfin supprimés et les phrases reconstituées par une linéarisation basée sur les feuilles (les mots), avec un tri des mots sur leur indice dans la phrase pour rétablir leur ordre original.

Ces trois étapes présentées de manière globale, nous pouvons à présent détailler les

deux outils utilisés : SYGFRAN, l'analyseur syntaxique, et COLIN, notre compresseur syntaxique de phrases.

4.3 Analyse syntaxique

Cette section a pour but de présenter SYGFRAN, l'analyseur syntaxique que nous avons choisi.

Il est consistué d'un programme pour le système opérationnel SYGMART, proposé dans [Chauché, 1984]. Nous commençons par présenter SYGMART, afin de disposer des bases nécessaires pour décrire SYGFRAN, aussi bien que pour décrire notre compresseur syntaxique, qui est aussi un programme pour SYGMART.

4.3.1 SYGMART : un outil de manipulation d'éléments structurés

SYGMART (Système Grammatical de Manipulation Algorithmique et Récursive de Texte) est un environnement opérationnel permettant la décomposition puis la transformation d'une chaîne textuelle en une structure arborescente, puis éventuellement, après les manipulations souhaitées sur la structure, la linéarisation de cette dernière pour réobtenir une chaîne textuelle. Pour réaliser ces différentes tâches, SYGMART est composé de trois modules : OPALE, TELESI[62] et AGATE. Chacun de ces modules exploite un ensemble de règles ainsi qu'un dictionnaire pour réaliser le traitement.

SYGFRAN est un ensemble de ce type qui vise à produire une analyse syntaxique des phrases du français. Nous illustrons ici le fonctionnement des modules de SYGMART à partir d'exemples issus des règles de SYGFRAN.

De nombreuses informations de cette section sont tirées du manuel de SYGMART, [Chauché, 2001].

4.3.1.1 Caractéristiques du modèle d'analyse syntaxique de SYGFRAN

4.3.1.2 OPALE : le module de décomposition morphologique

OPALE est le module qui prend en entrée la chaîne à analyser et en produit une décomposition morphologique. Son but est de définir une transition entre la chaîne textuelle en entrée et un élément structuré, de type arbre, en sortie. Cette transition est effectuée par un transducteur d'états finis construit sur la consultation d'un dictionnaire et la segmentation de la chaîne d'entrée, c'est-à-dire, pour SYGFRAN, la phrase à analyser.

[62] TELESI (Transduction d'ÉLÉment Structurés Indexés) est un nom très proche de *télésie*, qui est un nom provenant du grec et donné par Haüy aux trois gemmes les plus précieuses, le rubis, le saphir et la topaze d'Orient. L'opale et l'agate sont aussi des pierres précieuses, d'où les noms donnés aux deux autres modules par le concepteur de l'analyseur.

L'arbre de la figure 4.2 représente l'arbre (simplifié) produit par les règles OPALE de SYGFRAN, à partir de la phrase de l'exemple 4.1.

Exemple 4.1 *La branche flotte sur l'océan.*

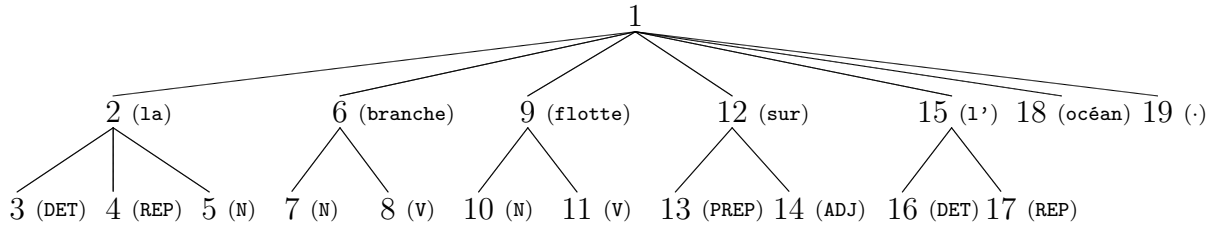

FIG. 4.2 – Exemple simplifié de sortie du module OPALE de SYGFRAN.

Nous utilisons ici une représentation graphique[63] similaire à celle de SYGFRAN pour le rendu de l'analyse. Les nœuds sont étiquetés par une numérotation effectuée sur un parcours en profondeur, utilisée pour référencer la liste des variables/valeurs associées à chaque nœud. Plutôt que de proposer une telle liste ici, nous préférons afficher, à côté de chaque nœud concerné, uniquement les variables qui sont pertinentes à cette section.

Lorsqu'il y a ambiguïté sur la catégorie lexicale d'un mot, un nœud est créé et se voit attribuer un fils par catégorie lexicale possible. C'est le cas pour les mots *la*, *branche*, *flotte*, *sur* et *l'* : *la* peut être un déterminant (`DET`), un pronom (`REP` pour représentant) ou un nom[64] (`N`) ; *branche* peut-être un nom ou un verbe (`V`), *idem* pour *flotte* ; *sur* peut être une préposition (`PREP`) ou un adjectif[65] (`ADJ`) ; enfin *l'* peut être un déterminant ou un pronom.

Cette ambiguïté ne peut être levée à l'analyse morphologique, qui ne considère pas les relations entre les mots. L'analyse syntaxique, dans le module TELESI, tentera d'effectuer cette tâche.

Pour les mots *la* et *flotte*, quelques-unes des variables renseignées sont présentées dans le tableau 4.1.

Les chaînes textuelles utilisées dans SYGFRAN pour définir les variables et leurs valeurs ne sont pas utilisées dans l'exemple. Elles sont remplacées par leur nom complet (ou abrégé) pour des raisons de clarté. Les noms de variables précédés du mot *sous* désignent la sous-catégorie du mot, ici *verbe conjugué*, *nom commun*, *article défini* ou *pronom personnel*. La variable *place mot* renseigne sur la position, en caractères, du mot dans le texte analysé. Elle sera utilisée pour ordonner les mots lors de la linéarisation de l'arbre élagué.

[63] Cette représentation est visualisable à partir de l'analyseur en ligne SYGFRAN, disponible sur le site professionnel de Jacques Chauché, `http://www.lirmm.fr/~chauche`

[64] Le nom *la* est la note de musique

[65] Dans le sens d'acide ou aigre. Par exemple *une pomme sure*.

var. \ mot	la			flotte	
nœud	3	4	5	10	11
flexion	la	la	la	flotte	flotte
lemme	le	le	la	flotter	flotte
catégorie	det.	pron.	nom	verbe	nom
genre	féminin	féminin	masculin	-	féminin
nombre	singulier	singulier	singulier	-	singulier
personne	-	-	-	1 | 3	-
mode	-	-	-	ind. | subj. | imp.	-
temps	-	-	-	présent	-
type	-	objet	-	trans. / intrans.	-
auxiliaire	-	-	-	avoir	-
sous verbe	-	-	-	verbe conjugué	-
sous nom	-	-	nom com.	-	nom com.
sous det.	art. def.	-	-	-	-
sous pron.	-	pron. pers.	-	-	-
place mot	0	0	0	11	11

TAB. 4.1 – Exemple d'informations renseignées par le module OPALE de SYGFRAN.

Plusieurs valeurs peuvent être spécifiées pour une même variable (séparées par le symbole « | » dans le tableau) lorsque la forme du mot ne permet pas de déterminer quelle est la réelle valeur dans la phrase. Par exemple, pour le mot *flotte* pris comme flexion du verbe *flotter*, il peut correspondre aussi bien à la première personne qu'à la troisième personne du singulier. L'ambiguïté devra être levée au niveau de TELESI.

Nous ne présentons pas les règles et le dictionnaire d'OPALE, car cela n'est pas utile à la description de notre approche. La seule interaction que nous avons eue avec ces éléments fut d'introduire quelques fois un mot ou une locution absente du dictionnaire, afin de rendre correcte l'analyse morphologique de certaines phrases puis leur analyse syntaxique. Ce qui importe ici c'est surtout structuration de la donnée en sortie d'OPALE.

4.3.1.3 TELESI : le module de transformation d'éléments structurés

TELESI est le module qui définit une transition entre des éléments structurés. Un élément structuré est un couple (E, S) où :

- E est un ensemble fini d'étiquettes ;
- S est un ensemble d'arborescences étiquetées (A_i, E_i, f_i) tel que $E_i \in E$;

avec A_i une arborescence et f_i une fonction qui associe chaque nœud de A_i à un et un seul élément de E_i.

Ainsi, TELESI peut manipuler simultanément plusieurs arborescences, toutes associées au même ensemble d'étiquettes. Chaque arborescence est considérée comme une dimen-

sion, ce qui fait de SYGMART un outil d'analyse multi-dimensionnel. Cependant, dans notre outil de compression nous ne manipulons qu'une seule arborescence, suffisante à notre travail. Notre présentation de ce module restera donc circonscrite à une seule dimension.

TELESI prend en entrée soit la sortie d'OPALE, soit la sortie d'un autre traitement TELESI, pour ensuite la transformer en la structure arborescente voulue. La transition est effectuée par un transducteur à pile composé simulant une grammaire transformationnelle.

Exemple de rendu. À titre d'illustration, l'arbre de la figure 4.3 est le résultat de l'application du module TELESI de SYGFRAN, à partir du précédent exemple de la figure 4.2.

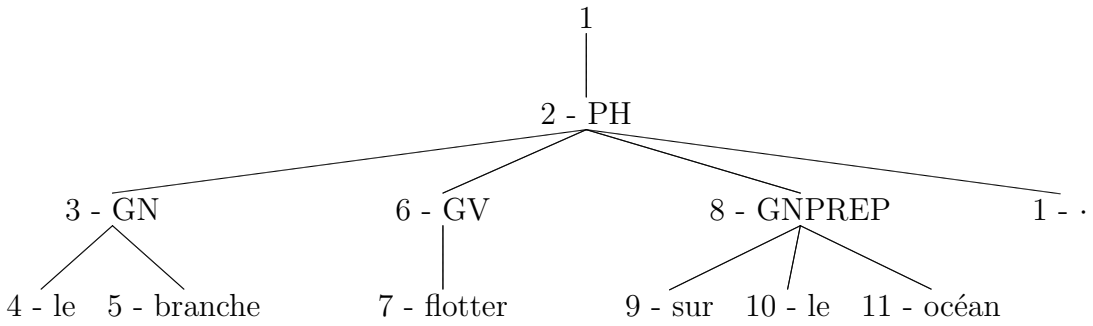

FIG. 4.3 – Exemple de sortie du module TELESI pour SYGFRAN.

Dans cet arbre sont ajoutés à la numérotation des nœuds :

- la catégorie lexicale du constituant, pour les nœuds internes (variable `K` dans SYGFRAN, qui prend ici comme valeurs : `PH` pour IP, `GN` pour NP, `GV` pour VP et `GNPREP` pour PP) ;
- le lemme, pour les feuilles (variable `LEMME` dans SYGFRAN).

Par la suite, nous n'utiliserons pas la numérotation comme étiquetage des nœuds lorsque ce ne sera pas indispensable, nous conserverons alors uniquement le lemme.

À noter ici les groupes nominaux prépositionnels (`GNPREP`) qui regroupent la préposition ainsi que les fils de son groupe nominal complément. Cela ne nous pose pas de problème car les compléments de la proposition ne sont jamais effaçables, il n'est donc pas question d'élaguer la branche de leur groupe nominal. Nous conserverons ce type d'attachement dans la suite de ce travail. Une autre différence vis-à-vis de nos règles structurelles est la suppression des nœuds internes portant la catégorie lexicale des mots, cette information se trouvant maintenant renseignée dans la variable `CAT`, associée à chaque feuille, l'information est donc toujours accessible.

Le réseau de grammaires. Les règles transformationnelles de COLIN se concentrent quasi-exclusivement sur le module TELESI[66], nous allons donc expliquer ici les bases fondamentales du fonctionnement des grammaires de TELESI.

Le système TELESI se compose d'un réseau conditionnel de grammaires élémentaires. Par réseau, nous entendons un ensemble de grammaires inter-connectées. L'application sur un élément structuré est définie par un cheminement dans ce réseau avec, en chaque point, une application d'une grammaire élémentaire. Ce cheminement est conditionnel et dépend de l'élément structuré initial. Certaines grammaires du réseau sont définies comme initiales et le cheminement débutera par l'une d'entre elles. Les sorties du réseau sont repérées par des marqueurs et le système définira une transition comme terminée seulement après avoir atteint un de ces marqueurs.

Chaque grammaire est définie par une liste ordonnée de règles transformationnelles et un mode d'application. Ce mode d'application permet une utilisation récursive du réseau, la modification des étiquettes associées aux racines ainsi qu'un parcours de l'élément structuré d'entrée. Toute transformation de l'élément structuré d'entrée est réalisée par une règle de transformation. Une grammaire élémentaire est donc un algorithme de Markov étendu aux éléments structurés et le réseau une composition simple ou récursive de ces algorithmes. Chaque règle d'une grammaire définit une transformation par remplacement d'un sous-élément structuré. Cette transformation s'effectue par la transformation simultanée des arborescences composant l'élément structuré. La définition d'une transformation s'effectue donc relativement à la définition d'un schéma d'élément structuré qui identifie une partie d'un élément structuré.

Exemple de grammaire TELESI. Afin de mieux saisir le fonctionnement, tout en introduisant certains éléments fondamentaux de la syntaxe, nous allons travailler sur un programme, une grammaire d'exemple, utilisant les variables de SYGFRAN, présentée dans le code source 4.1.

Cette grammaire est basique, elle n'exploite qu'un petit échantillon de la richesse syntaxique de SYGMART. Se référer au manuel de SYGMART pour davantage de détails [Chauché, 2001].

Nous considérons ici que cette grammaire vient se placer juste après celle du module TELESI de SYGFRAN, c'est-à-dire juste après l'analyse syntaxique. Notre grammaire va alors récupérer l'arbre enrichi produit par SYGFRAN, pour y effectuer un post-traitement. Ce n'est pas une grammaire de notre compresseur, mais juste un simple exemple qui effectue un traitement de même nature.

Le but de cette grammaire est :

[66] COLIN n'utilise pas le module OPALE, inutile pour notre compression syntaxique, cependant utilise le module AGATE pour linéariser la structure arborescente.

Code source 4.1 – Exemple de grammaire TELESI.

```
1  &REFER(VariablesAnalyseSyntaxique,GrammaireExemple).

3  &GRAMMAIRE.

5  &ENTREE: ATTACHE_COMPLEMENT(I).
   R_ATTACHE_GRIMPER:
7     0(1(2),3) /
         0:(K = PHRASE); 1:(K = GV); 2:(LEMME = 'grimper');
9        3:(K = GNPREP)&(SEMOBJ = LIEU)
      => 0(1(2,3)) /
11       3:3(FS = COMP).
   R_ATTACHE_INDEPENDAMMENT:
13    0(1(2),3(4)) /
         0:(K = GV); 1:(K = GADV); 2:(LEMME = 'indépendamment');
15       3:(K = GNPREP); 4:(CAT = PREP)&(LEMME = 'de')
      => 0(1(2,3(4))) /
17       3:3(FS = COMP).
   --> SUPPRIME_MODIFIEURS.
19
   &GRAM: SUPPRIME_MODIFIEURS(I).
21 R_SUPPRIME_MODIFIEURS:
      0(1) /
23       1:(FS = MOD)
      => 0.
25 --> %STOP.
```

1. de corriger un mauvais attachement de l'analyseur syntaxique pour
   - un complément du verbe *grimper*, de type locatif[67] ;
   - un complément de l'adverbe *indépendamment* ;
2. de supprimer les modifieurs de la phrase analysée.

Examinons-la, élément par élément.

Le mot clé `REFER`, ligne 1, spécifie deux informations. La première est le nom de l'ensemble de variables utilisé pour ce réseau de grammaires. Dans notre exemple, cet ensemble, nommé `VariablesAnalyseSyntaxique`, désigne celui de l'analyseur syntaxique SYGFRAN. Ces variables contiennent essentiellement des informations morphologiques et syntaxiques. Le tableau 4.2 fournit le nom long de chaque variable et valeur utilisée dans l'exemple.

Le seconde information est le nom de la grammaire, ici `GrammaireExemple`, utilisé comme référence pour l'exécution de cette grammaire dans le processus complet. Le mot clé `GRAMMAIRE`, ligne 3, indique le début du réseau de grammaires. Avant ce mot clé peuvent être placées des procédures conditionnelles ou d'affectation.

Le mot clé `ENTREE`, ligne 5, marque le début d'une grammaire d'entrée dans le réseau. Un réseau peut disposer de plusieurs entrées, cependant nous n'en utilisons qu'une dans l'exemple. Cette grammaire se nomme `ATTACHE_COMPLEMENT` et a pour but de corriger certains attachements de compléments. Plusieurs modes d'application des grammaires sont disponibles, dont :

[67] Nous considérons ici une acception du verbe grimper qui requiert un complément locatif.

Variable	Nom long	Valeur	Nom long
CAT	catégorie lexicale	PREP	préposition
FS	fonction syntaxique	COMP MOD	complément modifieur
K	catégorie du groupe (du constituant)	GADV GNPREP GV PHRASE	groupe adverbial groupe nominal prépositionnel groupe verbal proposition
LEMME	lemme		
SEMOBJ	sémantique de l'objet	LIEU	circonstant de lieu

Tab. 4.2 – Nom des variables et valeurs du code source 4.1.

- Itératif, notation I, mode indécidable (peut conduire à une application infinie de la grammaire élémentaire) où les règles de la grammaire sont testées et appliquées indéfiniment, jusqu'à ce qu'aucune ne puisse être appliquée ;
- Unitaire, notation U(n) ou U, où la liste des règles est testée et appliquée au maximum un nombre n fini de fois ;
- Exhaustif, où la grammaire est appliquée de façon itérative, mais à chaque pas d'itération, les règles appliquées sont éliminées pour les applications futures.

Ici nous avons choisi un traitement itératif pour les cas où plusieurs corrections similaires doivent être effectuées dans une même phrase.

Cette grammaire est composée de deux règles : R_ATTACHE_GRIMPER, ligne 6, et R_ATTACHE_INDEPENDAMMENT, ligne 12. Étudions maintenant le schéma de reconnaissance de ces deux règles.

La chaîne parenthésée de la ligne 7 correspond à la structure arborescente cherchée dans l'arbre enrichi. L'arbre de la figure 4.4 représente cet arbre recherché.

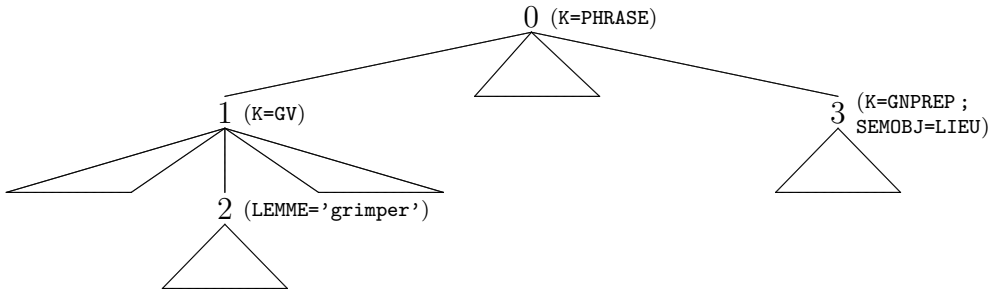

Fig. 4.4 – Arbre du schéma de reconnaissance de la règle R_ATTACHE_GRIMPER du code source 4.1 Telesi.

Chaque triangle signifie la présence potentielle d'un sous-arbre dans la structure recherchée. À noter que le nœud 0 peut aussi disposer de frères.

Les lignes 8 et 9 spécifient les contraintes de reconnaissance sur les variables des nœuds. Par exemple, la première contrainte spécifie que la variable `K` du nœud `0` doit avoir pour valeur `PHRASE`, autrement dit, la racine de l'arbre doit être un nœud de proposition. Une autre contrainte requiert que le groupe à attacher soit un groupe prépositionnel (`K=GNPREP`) de type locatif (`SEMOBJ=LIEU`), comme le veut la sous-catégorisation de l'acception du verbe *grimper* qui nous intéresse.

La règle `R_ATTACHE_GRIMPER` peut par exemple reconnaître le groupe verbal et le groupe prépositionnel de l'arbre syntagmatique de la figure 4.5, basé sur une analyse de la phrase *Félix grimpe avec agilité sur le toit*, où le complément du verbe *sur le toit* a été analysé, à tort, comme modifieur de la proposition. À noter que seules les variables utiles à la reconnaissance sont spécifiées dans l'arbre.

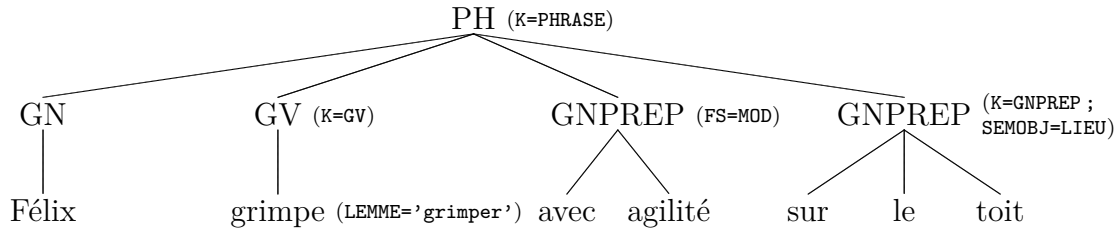

FIG. 4.5 – Exemple d'arbre identifié par le schéma de reconnaissance de l'arbre de la figure 4.4.

Dans cet exemple, le schéma de reconnaissance associe les nœuds ainsi : `0` ↔ `PH`, `1` ↔ `GV`, `2` ↔ `grimpe`, `3` ↔ `GNPREP`. Le groupe prépositionnel *avec agilité* n'empêche pas la reconnaissance de cet arbre, car il est compris dans le sous-arbre potentiel du nœud `0` du schéma de reconnaissance, entre les nœuds `1` et `3`. Il aurait été possible de l'en empêcher en ajoutant un astérisque comme frère droit du nœud `1` et frère gauche du nœud `3` comme ceci :

```
0(1(2),*,3)
```

Cependant, dans ce cas, nous souhaitons bien autoriser la présence de modifieurs de la proposition en cette position.

La structure une fois identifiée, la règle va procéder à sa transformation et/ou affectation de valeurs aux variables. Dans notre exemple, la règle effectue un attachement du sous-arbre de type groupe prépositionnel au groupe verbal. En effet, à la ligne 10, nous observons que le nœud `3` est maintenant fils du nœud `1`, qui est la racine du groupe verbal. De plus, la variable `FS` du nœud `3` se voit affecter, ligne 11, la valeur `COMP`, car le constituant prépositionnel est un complément. Dans la notation « `3 :3` », le premier `3` signifie que nous allons travailler sur l'ensemble des variables du nœud `3` de la structure transformée (ligne 10), et le second signifie que cet ensemble de variables est celui du

nœud 3 de la structure du schéma de reconnaissance (ligne 7). En d'autres termes, nous allons modifier les variables du nœud 3, plutôt que créer un nouveau nœud 3.

L'arbre transformé de notre exemple est représenté en figure 4.6.

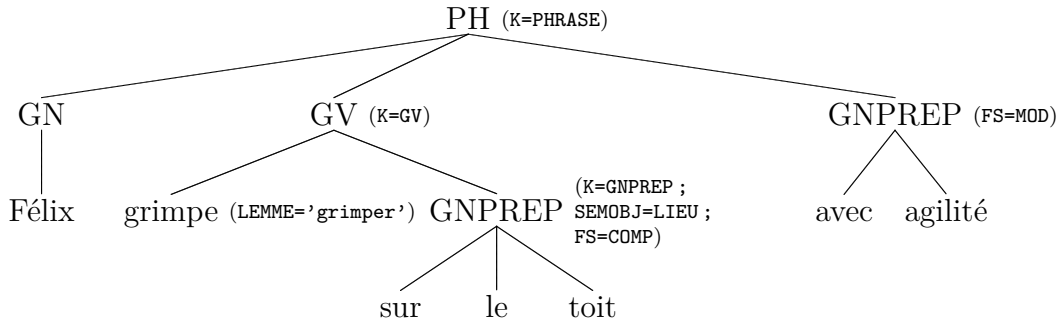

FIG. 4.6 – Arbre où l'attachement du complément du verbe de la figure 4.5 a été corrigé.

La seconde règle, R_ATTACHE_INDEPENDAMMENT, réalise un attachement similaire, mais pour un complément de l'adverbe. Elle impose la présence d'une préposition *de* dans le groupe prépositionnel recherché. Cette règle est par exemple applicable dans l'arbre de la figure 4.7, pour produire celui de la figure 4.8.

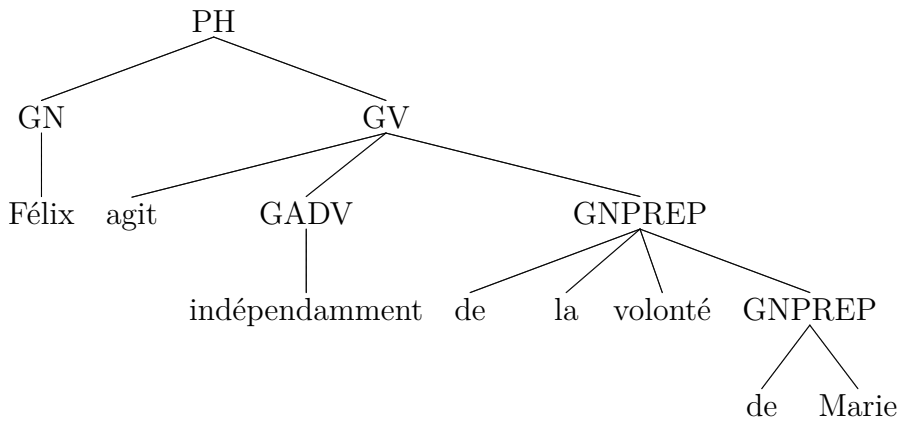

FIG. 4.7 – Exemple d'arbre identifié par la règle R_ATTACHE_INDEPENDAMMENT de la grammaire ATTACHE_COMPLEMENT du code source 4.1.

Enfin, la seconde grammaire, SUPPRIME_MODIFIEURS, ligne 20, possède une seule règle, R_SUPPRIME_MODIFIEURS, ligne 21, qui supprime systématiquement tous les constituants identifiés comme modifieurs. Elle recherche toute structure père – fils (ligne 22), où le fils a COMP comme valeur de la variable FS (ligne 23), pour ensuite supprimer ce fils (ligne 24).

Ainsi, sur l'arbre de la figure 4.6, lequel arrive en entrée à cette grammaire, le modifieur sera supprimé, produisant alors l'arbre de la figure 4.9.

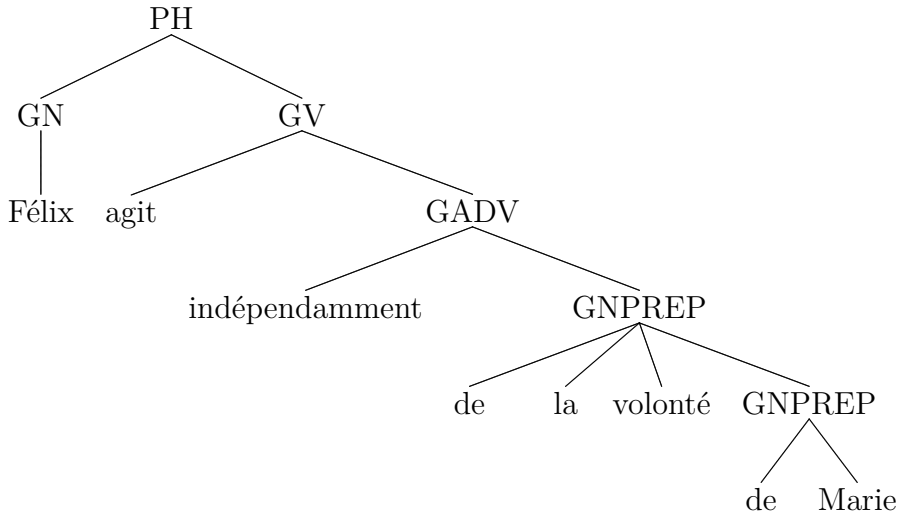

FIG. 4.8 – Arbre où l'attachement du complément de l'adverbe de la figure 4.7 a été corrigé.

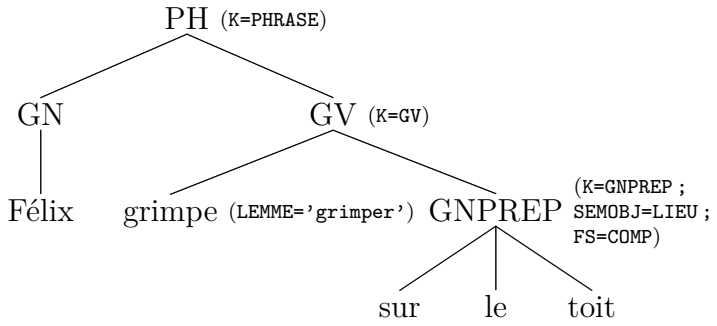

FIG. 4.9 – Arbre où le sous-arbre du modifieur a été élagué selon la règle `R_SUPPRIME_MODIFIEURS` du code source 4.1.

Nous venons de présenter les bases de la syntaxe des règles transformationnelles, utiles pour notre approche. Nous n'utilisons pas le dictionnaire TELESI dans nos règles de compression. Un tel dictionnaire peut être utilisé pour établir des liens spécifiques entre certaines valeurs de variables. Par exemple, le module TELESI de SYGFRAN utilise le dictionnaire principalement pour localiser les locutions et leur attribuer une catégorie lexicale.

Ainsi le module TELESI permet de transformer des structures, en déplaçant, supprimant ou ajoutant des branches, en réaffectant des variables ou en en proposant de nouvelles pour chaque nœud de la structure identifiée.

La puissance d'expression syntaxique de TELESI n'est qu'effleurée dans cette section. Ce module offre davantage de possibilités de reconnaissance structurelle ainsi que de transformation de la donnée, dont certaines sont utilisées dans les règles de notre compresseur. La présentation complète de SYGMART n'est cependant pas appropriée dans notre travail. Notre but ici est de présenter son fonctionnement général ainsi que de proposer une intro-

duction à la syntaxe. Le manuel de SYGMART regroupe toutes les informations suffisantes à la compréhension de notre compresseur.

4.3.1.4 AGATE : le module de linéarisation d'éléments structurés

Le but du langage AGATE est de définir une transition entre un élément structuré (un arbre) et une chaîne de caractères. Ce sont les ensembles d'étiquettes (couples variable – valeur) de chaque nœud de l'arbre qui sont exploités. Le transducteur sous-jacent du système AGATE est, comme pour le système OPALE, un transducteur d'états finis non déterministe. Il fournit la première solution possible (dans le système OPALE toutes les solutions sont prises en compte).

Nous ne présentons pas ici la syntaxe des règles d'AGATE (identique à celle d'OPALE), très peu utilisée dans notre système. En effet, dans notre cas, le mode de transition qui nous intéresse est un simple parcours des feuilles de l'arbre, en extrayant les formes fléchies de chaque mot (c'est-à-dire leur forme dans la phrase d'origine), pour enfin les mettre bout à bout, séparées par des espaces, et ainsi générer la phrase compressée. Quelques cas particuliers sont cependant pris en compte dans nos règles AGATE au sujet du placement des espaces, vis-à-vis des différentes ponctuations et de la typographie française.

4.3.2 SYGFRAN : l'analyseur syntaxique

Nous venons de présenter le système opérationnel SYGMART, dans lequel est exécuté le programme SYGFRAN, un ensemble de règles et de dictionnaires visant à produire une analyse syntaxique de phrases en français. Nous présentons maintenant SYGFRAN, en tant qu'analyseur syntaxique.

4.3.2.1 La grammaire de SYGFRAN

L'analyse syntaxique produite par SYGFRAN respecte assez rigoureusement nos règles structurelles définies dans le précédent chapitre. Il existe cependant quelques différences, dont certaines affectent notre approche. Nous présentons maintenant ces différences.

L'attachement des compléments. SYGFRAN adopte, pour certains constituants, un point de vue similaire à celui de la grammaire traditionnelle, où les verbes requérant un groupe circonstanciel comme compléments sont marqués d'un trait d'intransitivité. Ce comportement a pour conséquence de généralement attacher à la proposition ces constituants, comme pour le groupe *à Paris* dans la phrase *Je vais à Paris*, et donc de les considérer comme des modifieurs de la proposition, lesquels sont effaçables, à tort ici. D'autres attachements de compléments, notamment aux têtes adjectivales et adverbiales, ne sont pas toujours considérés dans SYGFRAN.

Comme dans le Le*fff*, ce sont les verbes qui disposent de l'information la plus complète sur leurs éléments sous-catégorisés. Cependant, le caractère obligatoire des compléments n'est pas spécifié dans SYGFRAN.

Ces différences ne sont dues qu'à une légère différence de point de vue dans les grammaires ainsi qu'une grande quantité d'informations de sous-catégorisation manquantes dans le dictionnaire de SYGFRAN. Ce dernier pourrait d'ailleurs être enrichi et ajusté sur ce point, à l'aide de ressources lexicales telles le Le*fff*.

Cette problématique a fait l'objet d'un stage de première année en informatique à l'École Normale Supérieure de Lyon[68]. Ce stage, intitulé « Exploitation de la construction syntaxique des verbes pour l'évaluation automatique de l'influence sémantique de leurs compléments », fut réalisé début 2006 par François Dupressoir, encadré par Augusta Mela, maître de conférences en sciences du langage à l'Université de Montpellier 3, et moi-même. Durant ce travail, F. Dupressoir a posé les prémisses à l'exploitation d'une ressource lexicale, le lexique-grammaire, dans l'objectif d'améliorer la qualité des attachements des compléments du verbe dans SYGFRAN, pour enfin prévenir la suppression des compléments obligatoires lors d'une compression de phrases.

Cette étude a abouti à une technique de correction de l'analyse qui consiste, dans les grandes lignes, à :

1. pour chaque verbe de la phrase, considérer l'ensemble de ses constituants frères et oncles dans l'arbre syntagmatique, c'est-à-dire ceux qui ont le plus de chance de correspondre à un complément du verbe mal attaché ;
2. pour chacun de ces constituants, sélectionner ceux qui se voient attribuer la même fonction syntaxique pour toutes les entrées, dans le lexique, du verbe considéré ;
3. pour enfin attacher les constituants sélectionnés à leur tête verbale.

Pour les constituants qui, selon les entrées du lexique, disposent de fonctions différentes vis-à-vis du verbe considéré, F. Dupressoir propose de marquer l'ambiguïté d'attachement en générant deux sous-arbres syntagmatiques dans le groupe verbal, un pour chaque attachement. Ainsi aucune information n'est perdue lors de l'analyse, et d'autres critères pourront être utilisés, dans un traitement ultérieur, pour trancher sur l'attachement.

Cette approche s'est concentrée sur les compléments du verbe, cependant le principe reste applicable à tout type de tête. En pratique, comme nous l'avons vu au chapitre précédent, les ressources lexicales exploitables adéquates à ce type de travail ne couvrent à l'heure actuelle qu'une petite partie du lexique. Nous n'avons donc pas encore pu réaliser une mise en œuvre de cette technique de correction sur l'attachement des compléments.

Il est cependant possible de procéder à des corrections manuelles par l'ajout de règles en post-traitement à l'analyse TELESI de SYGFRAN. À ce sujet, nous avons vu deux

[68] Site Internet : `http://www.ens-lyon.fr`

exemples de telles règles en section 4.3.1.3. Nous avons exploité cette technique pour garantir un bon attachement des compléments, sur un petit corpus de textes, lors de l'évaluation de COLIN, présentée dans le prochain chapitre. Ce procédé reste toutefois limité, car il n'est pas possible de créer manuellement des règles permettant de couvrir l'ensemble des sous-catégorisations pour tous les lemmes de toutes les catégories lexicales de têtes. De plus, une adaptation du dictionnaire et des règles de SYGFRAN, en intégrant ces informations de sous-catégorisation, est plus adéquate à cette problématique. Cela nécessite toutefois un travail d'enrichissement de grande envergure qui n'est envisageable qu'à plus long terme.

L'attachement des modifieurs. Certains modifieurs ne révèlent que peu d'information sur l'élément auquel ils doivent s'attacher. Parfois, des connaissances très précises sur les têtes impliquées sont nécessaires. Ainsi dans l'exemple 4.2, le constituant souligné de la phrase *a* est modifieur du nom *salade*, alors que dans la phrase *b*, le constituant souligné est modifieur de la proposition.

Exemple 4.2

a *Je mange une salade* <u>*avec des croûtons*</u>*.*

b *Je mange une salade* <u>*avec des baguettes*</u>*.*

SYGFRAN les analyse tous deux comme attachés à la proposition. Pour déterminer l'attachement correct dans la phrase *a*, il faudrait disposer de l'information que des croûtons ne peuvent habituellement être utilisés comme instrument pour le prédicat *manger*.

D'autres constituants ne fournissent aucune information sur leur attachement, révélant alors une réelle ambiguïté. Ainsi dans l'exemple 4.3, il est impossible de déterminer si les jumelles sont l'instrument avec lequel le sujet regarde la fille ou si ces jumelles sont un objet que la fille porte (ou des êtres humains jumeaux de sexe féminin accompagnant la fille) : les deux attachements sont possibles et tout à fait corrects.

Exemple 4.3 *Je regarde une fille* <u>*avec des jumelles*</u>*.*

Comme pour le précédent cas, SYGFRAN attache par défaut le constituant souligné à la proposition.

Ces problèmes d'attachement affectent moins la qualité des compressions produites, puisque la fonction syntaxique de ces constituants mal attachés reste celle de modifieur.

Les variables syntaxiques. Un certain nombre d'autres différences porte sur la terminologie des fonctions syntaxiques. SYGFRAN utilise un vocabulaire inspiré de grammaires telles le Grevisse ou le Bescherelle, qui diffère sensiblement du nôtre. Ainsi, un adjoint dans SYGFRAN désigne un modifieur. Au sujet des compléments, un attribut du sujet ou un complément d'objet direct, indirect ou second désigne un complément du verbe. SYGFRAN attribue la fonction de sujet au groupe sujet, laquelle correspond à un complément obligatoire dans notre grammaire. Il nous suffit alors de rechercher les sujets et de leur attribuer notre fonction. Enfin, SYGFRAN n'attribue pas de fonction au prédicat. Cependant, le groupe verbal complément de la proposition est facilement identifiable, car c'est le seul fils du nœud propositionnel de type groupe verbal fléchi.

Cette terminologie est utilisée au niveau des variables renseignées pour chaque nœud de l'arbre. Afin de retrouver notre terminologie, nous avons créé notre propre variable fonctionnelle, qui exploite alors les variables précédemment citées pour déterminer la fonction syntaxique de nos constituants.

4.3.2.2 La couverture syntaxique de SYGFRAN.

Nous terminons la présentation de SYGFRAN par un élément peu négligeable : la qualité de ses résultats. En effet, de cette donnée va directement dépendre la qualité des compressions de phrases de COLIN. Nous considérons ici la qualité d'analyse vis-à-vis des règles de SYGFRAN, et non pas vis-à-vis de notre grammaire, ce dont nous avons discuté dans en section 4.3.2.1.

Syntaxe ambiguë. Parfois, plusieurs interprétations syntaxiques peuvent coexister dans une même phrase, lorsque qu'elle valide plusieurs schémas de règles différents. Voici quelques phrases qui vérifient de telles propriétés :

- la petite brise la glace ;
- le pilote ferme la porte ;
- il creuse la route et l'asphalte.

La politique de SYGFRAN est de conserver cette ambiguïté en dédoublant l'arbre syntagmatique au niveau d'un nœud ajouté, ayant `PHAMBG` (phrase ambiguë) pour valeur de la variable `LEMME`. Ainsi, pour la première phrase d'exemple, SYGFRAN retourne l'arbre de la figure 4.10.

Dans le cadre de notre compresseur de phrases, nous ne disposons pas d'information pour déterminer l'attachement correct, dépendant du contexte, nous avons donc choisi de faire le choix arbitraire de conserver uniquement le sous-arbre gauche.

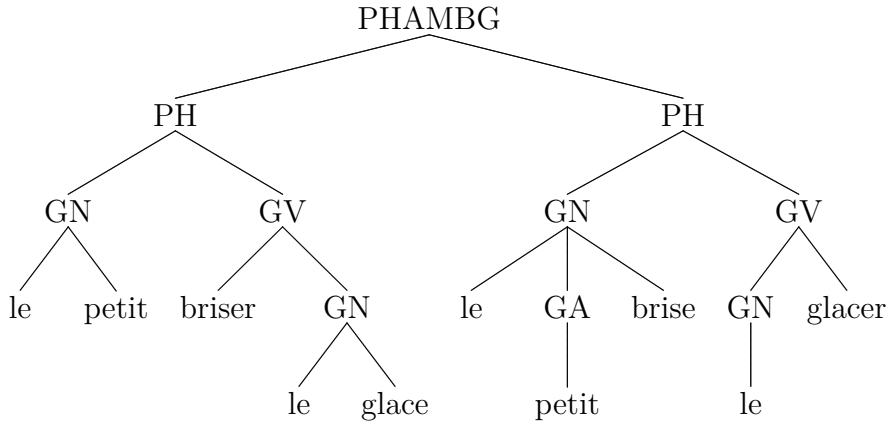

FIG. 4.10 – Exemple de construction syntaxique ambiguë analysée par SYGFRAN.

Analyse partielle. Un échec d'analyse intervient soit lorsque SYGFRAN ne reconnaît pas la catégorie lexicale de certains mots de la phrase, soit lorsque l'analyseur ne dispose pas de schéma structurel reconnaissant la disposition de certains mots ou constituants de la phrase. SYGFRAN retourne alors un arbre aussi complet qu'il peut fournir, tout en attachant à la racine les mots inconnus et les constituants dont la fonction syntaxique n'a pu être déterminée. La délimitation des constituants est alors souvent erronée. La variable `LEMME` du nœud racine de la phrase prend alors la valeur `ULFRA`, pour unité linguistique française (inconnue ici), plutôt que `PH`, pour une phrase bien analysée.

Si un mot n'est pas dans le dictionnaire, SYGFRAN ne peut pas connaître sa catégorie lexicale. Il peut cependant tenter de la déduire, si la place du mot dans la structure de la phrase implique, sans ambiguïté, sa catégorie. Par exemple, dans la phrase *le gremlin mange*, SYGFRAN ne connaît pas le nom *gremlin*, indiqué par une valeur `INCONNU` pour la variable `CAT`, mais parvient tout de même à déduire que *le gremlin* est un groupe nominal. L'arbre de la figure 4.11 représente une telle analyse.

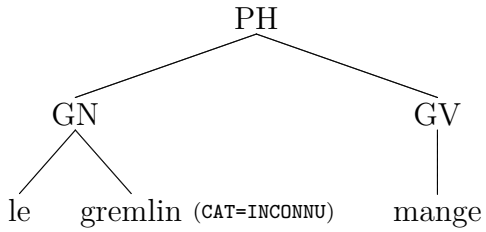

FIG. 4.11 – Exemple de catégorie lexicale déduite par SYGFRAN.

Par contre, dans la phrase *c'est le gremlin qi mange une pomme*, la faute d'orthographe sur le pronom *qui* produit une erreur dans l'analyse, SYGFRAN se retrouvant incapable d'identifier le pronom.

L'arbre de la figure 4.12 illustre ce cas.

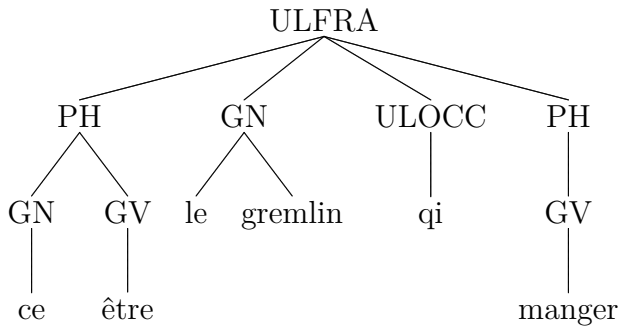

FIG. 4.12 – Exemple d'analyse partielle de SYGFRAN causée par une catégorie lexicale inconnue.

Le mot dont la catégorie lexicale n'a pas pu être déterminée, reste attaché à la racine de la phrase par un nœud de type groupe à catégorie indéterminée, appelé `ULOCC` (pour *Unknown Locution*). Une telle indétermination peut aussi se produire si aucune des catégories lexicales du mot spécifiées dans le dictionnaire ne peut être associée à une des constructions syntaxiques possibles pour la phrase analysée, selon les règles TELESI de SYGFRAN. Nous verrons un tel cas dans la figure 4.13.

Les constituants dont SYGFRAN ne parvient pas à déterminer la fonction syntaxique sont attachés au sommet de la phrase. Cela n'empêche pas une analyse correcte de leurs sous-constituants.

Par exemple, la phrase *Félix mange et court plus vite que la moyenne*, met en difficulté[69] SYGFRAN, qui produit deux analyses partielles, dont la première est présentée en figure 4.13.

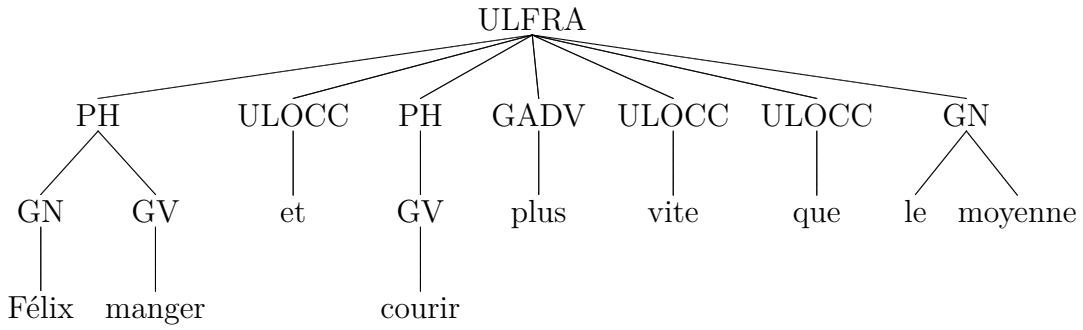

FIG. 4.13 – Exemple d'analyse partielle de SYGFRAN causée par une construction syntaxique inconnue.

[69]L'analyseur évoluant mois après mois, cette construction pourrait ne plus être une difficulté pour SYGFRAN à l'heure où vous lisez ces lignes.

Nous pouvons noter que SYGFRAN a identifié correctement le premier groupe sujet (*Félix*) et verbe (*manger*), ainsi que le dernier groupe nominal *la moyenne*. Cependant, il n'est pas parvenu à disposer tous les mots et constituants entre eux.

L'impact d'une analyse partielle sur la compression de phrases est assez important. D'une part, les constituants dont la fonction syntaxique n'est pas identifiée ne pourront faire l'objet d'une suppression, et d'autre part les constituants dont la délimitation a échoué ne peuvent être supprimés sans risquer fortement de causer une agrammaticalité. Ainsi, dès que SYGFRAN retourne un nœud étiqueté `ULFRA`, il est fortement conseillé de ne pas supprimer tous ses nœuds fils. Cela a pour conséquence de faire chuter considérablement le taux de compression.

4.3.2.3 Caractéristiques techniques de SYGFRAN

Modèle d'analyse syntaxique. SYGMART, et donc SYGFRAN, fonctionne dans un mode d'analyse de type calculatoire, par opposition au classique mode combinatoire, selon la classification des modèles d'analyse syntaxiques définis par Jacques Vergne dans [Vergne, 2001]. Certaines caractéristiques de SYGMART le démarquent toutefois de la description du mode calculatoire classique présentée dans cet article :

- les ressources lexicales ne se limitent pas à « des mots grammaticaux, des morphèmes de fin de mots » mais comprennent un lexique généré exhaustif ;
- le processus de calcul exploite bien un ensemble de règles « conditions ⇒ actions », cependant le moteur n'effectue pas une passe « une fois sur chaque unité » textuelle considérée, mais de une à plusieurs fois, selon le mode de traitement de la grammaire élémentaire (unitaire, exhaustif, itératif ou récursif, comme vu en section 4.3.1.3), le cheminement conditionnel dans le réseau de grammaire et l'applicabilité des règles concernées ;
- le traitement de la donnée ne se fait pas à travers un « flux à débit constant », où « un élément du flux est traité complètement, une fois pour toutes, en passe unique, avant de passer à l'élément suivant », mais par un traitement global du texte, où une règle peut s'appliquer sur n'importe quel élément, SYGMART n'imposant aucune contrainte d'ordre de traitement des éléments textuels ;
- le lexique du texte analysé n'est calculé et produit en sortie que pour les items lexicaux non présents dans le dictionnaire d'OPALE. Pour ces derniers, la catégorie grammaticale est inférée en fonction de la structure syntaxique détectée la plus probable ;
- enfin, la complexité du processus d'analyse est en $O(n.log_2(n))$ plutôt que linéaire (voir le détail en section 4.3.2.3).

critères	*compilation*	*analyse de langues combinatoire*	*tagging, chunking*	*analyse de langues calculatoire*	SYGFRAN
modèle des structures	grammaire formelle	grammaire formelle	aucun	aucun	aucun
ressources syntaxiques	exhaustives (grammaire formelle)	exhaustives (grammaire formelle)	partielles (règles contextuelles)	règles : condition ⇒ action	règles contextuelles : condition ⇒ action
ressources lexicales	exhaustives (primitives)	exhaustives (dictionnaire)	exhaustives ou partielles	mots grammaticaux seulement	exhaustives ou partielles[a]
processus	répétitif / token, déterministe	arborescent, combinatoire, non déterministe	répétitif / token, calculatoire, déterministe	répétitif / token, calculatoire, déterministe	répétitif / token, calculatoire, déterministe
complexité en temps	*théorique :* polynomiale, *pratique :* linéaire	*théorique :* exponentielle, *pratique :* polynomiale	*théorique :* linéaire, *pratique :* linéaire	*théorique :* linéaire, *pratique :* linéaire	*théorique :* $O(n.log_2(n))$, *pratique :* $O(n.log_2(n))$
code analysé	langage formel	langue	langue	langue	langue

TAB. 4.3 – Comparaison des quatre modèles d'analyse syntaxique présentés dans [Vergne, 2001] avec celui de SYGFRAN.

[a]Inférence sur la catégorie grammaticale pour les items lexicaux absents du dictionnaire.

Afin de situer plus clairement le modèle d'analyse de SYGFRAN ainsi que ses particularités, nous reprenons maintenant le tableau, de Jacques Vergne, synthétisant les caractéristiques propres à chaque modèle d'analyse, en y ajoutant une colonne pour SYGFRAN, produisant ainsi le tableau 4.3.

Volume d'informations. L'analyse syntaxique s'appuie sur un grand nombre de règles, regroupées en grammaires pour celles du module TELESI, et d'entrées de dictionnaire pour les modules OPALE et TELESI, toutes créées manuellement. Le tableau 4.4 présente le nombre (arrondi) de ces règles, grammaires et entrées à l'heure où sont écrites ces lignes.

	nb. de règles	nb. de grammaires	nb. d'entrées dans le dictonnaire
OPALE	500	-	23500
TELESI	17200	250	1200

TAB. 4.4 – Volume d'informations dans SYGFRAN.

Il est important de considérer le fait que le dictionnaire OPALE de SYGFRAN est un lexique généré, c'est-à-dire qu'il ne présente pas toutes les flexions de chaque lemme mais uniquement une forme canonique (masculin singulier pour les noms et adjectifs et infinitif pour les verbes) et se base sur leur racine et les suffixes de flexion pour identifier tous

les éléments fléchis de la phrase. D'autres informations permettent de couvrir davantage l'ensemble des flexions identifiables dans la phrase, comme les traits de substantivation des adjectifs et verbes[70]. Ainsi le nombre d'entrées du dictionnaire n'est pas comparable à d'autres comme le Le*fff* qui proposent une entrée par flexion multipliant alors considérablement la taille de leur ressource.

Complexité de l'analyse. Comme nous venons de le voir, SYGMART doit manipuler une grande quantité de règles transformationnelles pour produire une analyse syntaxique des phrases. La complexité du processus de traitement est primordiale lorsqu'il s'agit d'analyser de gros volumes de données.

L'analyse morphologique (module OPALE) se réalise en complexité linéaire. La recherche de l'applicabilité d'un ensemble de règles (dans une grammaire élémentaire) sur la structure arborescente (module TELESI) se réalise en complexité linéaire. L'application d'une règle modifie la structure arborescente. Les nœuds de cette structure sont rarement unaires, peu souvent binaires et généralement ternaires. Chaque application construit au moins une hauteur d'arbre. Sachant que la largeur de l'arbre est de l'ordre du nombre de mots du texte (n), et en prenant en compte de l'arité moyenne de chaque nœud, une borne maximum fiable de complexité d'application d'une règle est alors de $log_2(n)$.

Ainsi, pour un nombre de règles égal à k et une donnée exprimée en nombre de mots n, SYGMART dispose d'une complexité théorique d'analyse qui est en $O(k * n * log_2(n))$. De plus, il s'agit d'une limite supérieure, car l'analyseur étant structuré en plusieurs grammaires ordonnées, le facteur multiplicatif réel est beaucoup plus petit que k (nous l'avons estimé à environ 16). Cela dit, même ainsi, plus le texte est important, plus k est petit devant n. Aujourd'hui SYGFRAN analyse un corpus de 220000 phrases, d'en moyenne 25 mots, en environ 24 heures, sur un ordinateur grand public disposant d'un processeur cadencé à 2,4 Ghz et d'une capacité de mémoire vive de 1 Go. Pour un texte d'environ 1000 mots, l'analyse prend environ 8 secondes, sur la même machine.

Lors de la construction de nos règles de compression, cette puissance fut fort appréciable, car elle nous a permis d'obtenir une compression dans un temps très court et donc des ajustements et enrichissements rapides des règles. L'évaluation de COLIN a aussi profité de cette puissance, pour la partie où une interaction avec un utilisateur intervient, proposant alors un résultat rapide à ce dernier et évitant de présauvegarder l'analyse des corpus d'évaluation.

Pourcentage de couverture syntaxique. Les cas d'analyses partielles de phrases représentent, en août 2006, environ 65 % de l'ensemble des analyses d'un corpus de 280000

[70]Par exemple pour le verbe *poster*, sont précisées les possibilités de substantivation en *poste* ou *postage* à partir de la racine *post* et des suffixes flexionnels *e* et *age*.

phrases extraites de corpus de documents variés, sensés représenter les genres de texte et les cas de syntaxes les plus courants. Ce résultat encourageant[71] fut confirmé lors de la campagne d'évaluation EASY, où l'analyseur a obtenu des résultats extrêmement honorables.

4.4 Le compresseur de phrases CoLin

Dans cette section nous présentons le compresseur de phrase CoLin[72] (COmpresseur LINguistique) que nous avons développé à partir de notre étude théorique du chapitre précédent. Il n'intègre pas tous les critères de sélection des constituants à supprimer, car certains ne sont que faiblement exploitables à l'heure actuelle, notamment les traits de sous-catégorisation portant sur le caractère obligatoire ou non des compléments, ainsi que les fonctions lexicales.

Nous faisons l'hypothèse, dans cette section, que l'analyse syntaxique de Sygfran est correcte afin de disposer des informations nécessaires à notre approche. Il fut un temps où nous tentâmes une approche visant à produire une compression à partir d'analyses partielles de Sygfran, à l'aide d'heuristiques prenant en compte les résultats dégradés, cependant les compressions produites voyaient leur qualité et taux de compression décroître dans des proportions telles que le résumé n'était plus significatif. Pour cette raison, lorsque nous avons évalué notre compresseur, nous avons préféré effectuer un post-traitement à l'analyse de notre corpus d'évaluation, afin d'obtenir une analyse correcte de l'ensemble de ses phrases.

Comme nous l'avons expliqué dans la précédente section, CoLin est un programme pour Sygmart. Cependant, ce ne fut pas toujours le cas. La première version du prototype fut écrite en langage Java. La raison est que nous disposions d'un ensemble de classes Java permettant de gérer plusieurs tâches en TALN par une architecture multi-agent, issues du système Blexisma, développé par Didier Schwab, [Schwab, 2005]. L'un des agents a pour tâche la récupération et l'analyse de l'arbre enrichi produit par Sygfran. C'est une classe Java qui implémente un agent chargé :

- de questionner le serveur de Sygmart, en fonction d'un texte fourni en entrée par l'utilisateur ;
- puis de récupérer le résultat de Sygfran, l'arbre enrichi ;
- pour enfin le convertir en une structure interne de type arbre morpho-syntaxique, qui pourra ensuite être manipulée facilement par le programme Java.

[71] Les analyseurs syntaxiques ont en général des taux d'échec plus importants, en raison de la complexité des instructions, de l'agrammaticalité de nombreux corpus, et de la présence de mots ou de tournures inconnus.

[72] Accessible en ligne depuis mon site professionnel : `http://www.lirmm.fr/~yousfi`

Mon travail consistait alors à analyser cette structure pour localiser des schémas vérifiant des contraintes sur la structure et les valeurs. Chaque groupe de contraintes constituait une règle de sélection des constituants.

La partie théorique progressant, les règles de sélection des constituants se sont petit à petit étoffées, et la reconnaissance des schémas s'est vue progressivement devenir de plus en plus complexe. Java ne disposant pas d'un système permettant de reconnaître et transformer facilement des éléments structurés en arborescence, nous nous sommes orientés vers le système opérationnel SYGMART, disposant d'un langage totalement adéquat à cette tâche. Nous avons alors recodé notre outil de sélection des constituants dans ce langage.

Un autre avantage de ce choix est d'éviter de devoir traduire la structure de données sortante de SYGFRAN, car nous utilisons maintenant le même format. Notre compresseur s'insère alors à la suite du flux de données géré par SYGMART, sous la forme d'un ensemble de réseaux de grammaires composées de règles transformationnelles.

Enfin, afin de faciliter l'utilisation de COLIN, pour nous comme pour l'utilisateur, nous avons développé une interface Web à notre outil, qui permet de compresser un texte en mode automatique ou semi-automatique, c'est-à-dire grâce à une interaction avec l'utilisateur. Pour permettre cette possibilité d'interaction, nous avons fait le choix de ne pas produire la compression directement dans nos règles transformationnelles, mais plutôt d'étiqueter les constituants du texte potentiellement effaçables d'informations sur les traits linguistiques pertinents à notre compression, pour ainsi donner à l'utilisateur la possibilité de choisir quels constituants seront effacés, à travers une interface, pour un résumé semi-automatique, ou selon le paramétrage préétabli pour la tâche visée, pour un résumé automatique.

La figure 4.14 représente le schéma de déploiement de COLIN, incluant l'utilisateur.

Le texte est saisi par l'utilisateur par l'intermédiaire d'un champ texte proposé par l'interface Web, dans son navigateur Web. Cette dernière envoie la donnée au logiciel serveur d'interface avec l'outil SYGMART, situé sur l'ordinateur serveur de SYGMART et qui est chargé de questionner l'analyseur pour obtenir le texte étiqueté qui sera retourné à l'interface Web. Enfin, l'interface Web propose à l'utilisateur un mécanisme d'interaction pour manipuler le texte étiqueté et produire la compression de texte souhaitée.

Nous présentons maintenant les règles transformationnelles de COLIN, section 4.4.1, puis son interface Web, section 4.4.2.

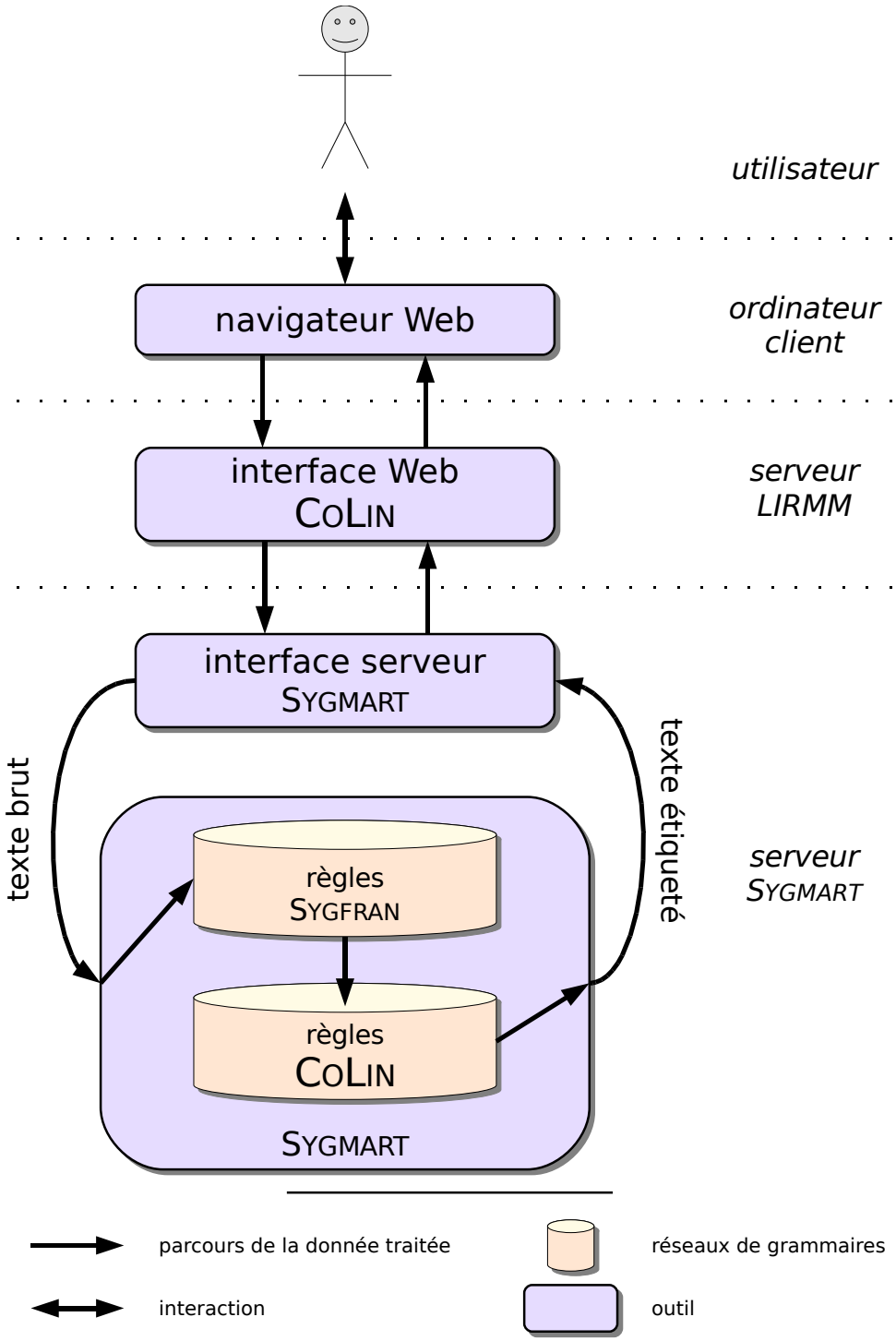

Fig. 4.14 – Schéma de déploiement de CoLin.

4.4.1 Les règles de compression de CoLin

Les règles s'organisent en plusieurs réseaux de grammaires, chacun disposant d'un rôle particulier dans le traitement. Le schéma de la figure 4.15 décrit l'enchaînement des réseaux de grammaires de Sygfran et CoLin dans Sygmart.

Les deux premiers réseaux concernent des grammaires de Sygfran, une pour l'analyse morphologique (module Opale), la seconde pour l'analyse syntaxique (module Telesi), puis la donnée structurée est transmise à l'ensemble de réseaux de grammaires de CoLin.

Nous les présentons maintenant un à un, en décrivant le fonctionnement global. Pour davantage de détails sur ces grammaires et règles, se reporter aux annexes où la majeure partie code source, en langage Telesi, est disponible.

4.4.1.1 Grammaires de post-traitement à Sygmart

Son but est de préparer l'arbre enrichi à notre sélection des constituants à supprimer. Nous la décrivons maintenant partie par partie.

Choix arbitraires. Lorsque Sygfran a décelé une ambiguïté et qu'il l'a représentée par un dédoublage l'arbre syntagmatique comme nous l'avons vu en section 4.3.2.2, cette première partie choisit arbitrairement un des sous-arbres. Par exemple, lorsqu'il y a une ambiguïté syntaxique, nous conservons ici le sous-arbre de gauche, qui correspond à la première solution proposée. Nous procédons ainsi car nous n'avons pas, dans ce contexte, les moyens de résoudre l'ambiguïté.

Corrections de l'analyse. Cette partie regroupe un ensemble de grammaires de corrections portant sur la structure ainsi que sur les variables de l'arbre enrichi. Par exemple, certains constituants gouvernés par une tête n'ont pas de fonction syntaxique par rapport à cette tête, nous leur donnons alors la fonction la plus probable.

Construction syntaxique des verbes. Nous définissons ici certains traits de sous-catégorisation pour un ensemble de verbes, qui ont été rencontrés dans le corpus d'évaluation de CoLin. Ces règles recherchent des constituants attachés à la proposition mais qui devraient l'être au groupe verbal, à partir de leur catégorie (groupe prépositionnel, proposition subordonnée, proposition infinitive...), et d'autres informations, comme la préposition utilisée (pour les `GNPREP`) ou le type de circonstant (lieu, temps...). Nous avons vu de telles règles, simplifiées, dans l'exemple de programme Telesi en section 4.3.1.3. Le but de ces règles est d'améliorer la qualité de l'analyse syntaxique afin de valider plus rigoureusement notre approche lors de son évaluation, qui est présentée dans le prochain chapitre.

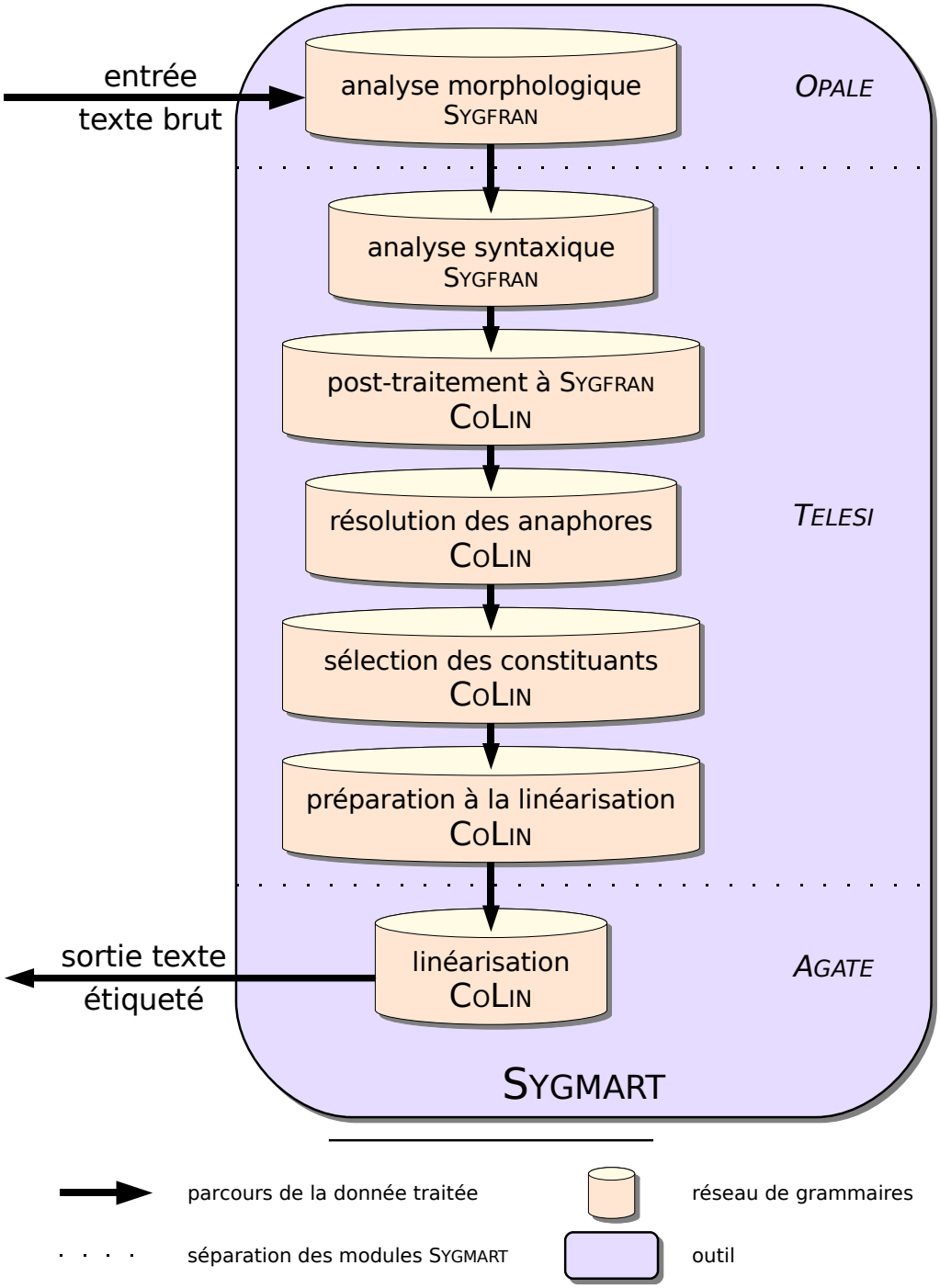

FIG. 4.15 – Les réseaux de grammaires de COLIN.

Gestion des formes contractées et composées. Le module OPALE de SYGFRAN décompose les formes contractées, comme les articles définis contractés (*à le* pour *au*, *de le* pour *du*...), lors de l'analyse syntaxique. Afin de restituer la contraction dans la phrase compressée après la linéarisation de l'arbre, nous effaçons la version décomposée de l'arbre.

Le même module rétablit à l'infinitif les verbes aux temps composés. Or, toujours dans le même souci de finalité, il est important pour notre outil de récupérer la forme composée. Nous effectuons alors le processus inverse qui éclate le verbe à l'infinitif, en sa forme composée.

4.4.1.2 Grammaires de résolution des anaphores

Afin de prévenir certaines suppressions de constituants référés par des pronoms, nous avons créé un réseau de grammaires qui vise à localiser les liens anaphoriques les plus probables, selon des critères classiques de correspondance en genre et en nombre des pronoms et groupes nominaux.

Par exemple, dans la phrase *a* de l'exemple 4.4, le modifieur de phrase *sur un toit* ne doit pas être supprimé, car le pronom *il* s'y réfère. Sa suppression, obtenue en phrase *b*, engendrerait une mauvaise compréhension de la phrase.

Exemple 4.4

a Félix court sur un toit, celui-ci est glissant.

b Félix court, celui-ci est glissant.

Notre algorithme commence par marquer, pour chaque pronom de chaque phrase, tous les groupes nominaux le précédant dans la phrase ou dans la précédente phrase. Puis, parmi ces groupes marqués, est étiqueté comme lié au pronom celui qui en partage le genre et le nombre, et qui lui en est le plus proche. Les règles parcourent tous les constituants et sous-constituants pour effectuer cette tâche. Pour les pronoms possessifs, nous n'effectuons pas de test sur le genre, car le pronom ne porte pas le genre de l'entité à laquelle il se réfère.

4.4.1.3 Grammaires de sélection des constituants

Elles ont pour but d'étiqueter les nœuds des constituants susceptibles d'êtres supprimés et s'appuient sur l'étude du chapitre précédent ainsi que sur la grammaire de SYGFRAN, vue en section 4.3.2.1.

Définition des modifieurs et compléments. La première étape et de définir les fonctions des constituants gouvernés, selon notre grammaire. Cela consiste juste en un renommage de la terminologie de SYGFRAN, comme vu en section 4.3.2.1. La fonction syntaxique de SYGFRAN étant sensiblement différente de la nôtre, nous avons créé notre propre variable de fonction pour effectuer cette définition.

Sélection des modifieurs et compléments. Ensuite sont sélectionnés, tête par tête, les constituants qui ont une chance d'être supprimés dans la phase finale de COLIN. Cette partie est circonscrite aux informations fournies par SYGFRAN. Ainsi sont modifieurs les constituants déterminés comme compléments circonstanciels et les adjoints. Les compléments sont les compléments d'objets du verbe, les groupes prépositionnels non-objets du verbe (la préposition est assez fiable comme propriété du complément du nom, de l'adjectif et de l'adverbe) ainsi que les compléments d'agent.

Ici sont aussi différenciés les modifieurs circonstants de lieu, de temps et les autres, car cette information est fournie par SYGFRAN. Ainsi il nous est possible de tester séparément leur importance. Pour les modifieurs du nom, une distinction est effectuée pour les détachés, lesquels sont généralement moins importants. Les modifieurs de négation ne sont quant à eux jamais sélectionnés.

Une variable (`SUPPRTYPE`) a été ajoutée afin de renseigner chaque constituant concerné sur quel type de suppression il est sujet, selon son importance. Par exemple, une valeur de modifieur de nom apposé peut être attribuée.

Verrous sur l'effacement. Il existe plusieurs cas particuliers où les modifieurs ou compléments ne peuvent généralement être effacés. Ces grammaires gèrent les cas suivants :

- les constituants verrouillés par un lien anaphorique ;
- les modifieurs du nom de type attributs du sujet, donc compléments obligatoires, y compris pour les circonstants[73] ;
- les modifieurs de la proposition impliqués dans une forme interrogative ;
- les modifieurs du nom précédés d'un article défini.

Une variable de verrou (`VERROU`) a été ajoutée pour cet usage, laquelle renseigne quel cas de verrou s'appliquent au constituant concerné.

Ensuite, les verrous de type lien anaphorique sont appliqués sur les antécédents des constituants concernés, afin d'éviter de perdre le lien suite à une suppression d'un constituant englobant. Enfin, les verrous fiables sont appliqués, c'est-à-dire qu'ils préviennent strictement l'effacement des constituants concernés. Pour cela, en pratique, la variable `SUPPRTYPE` est remise à vide. À l'heure actuelle, ce sont les compléments du verbe de type attribut du sujet qui sont sujets à ce type de verrou.

[73] Par exemple pour le complément du verbe dans la phrase *Félix est sur le toit*.

4.4.1.4 Grammaires de préparation à la linéarisation

Leur but est de préparer le rendu linéaire final. Le module AGATE parcourt les feuilles de l'arbre pour produire une chaîne de caractères, à partir de la variable FRM de SYGFRAN, qui contient la forme d'entrée du module OPALE, c'est-à-dire le mot fléchi, et donc celui qui doit être restitué dans la chaîne de sortie de COLIN. AGATE ne peut cependant pas effectuer des manipulations comme le tri des mots suivant la valeur d'une variable ou l'étiquetage de constituants, lesquels requièrent un traitement sur les nœuds internes de la structure, ce que nous faisons ici.

Nous avons choisi, comme étiquetage, de placer des balises XML autour des constituants sélectionnés. Ces balises contiennent, sous forme d'attributs XML, le type de constituant à supprimer ainsi que d'autres informations utiles au traitement final de l'interface Web. Dans l'arbre, elles sont placées comme feuilles, en frère gauche et frère droit des nœuds concernés. La chaîne de la balise est placée dans la variable FRM, ainsi elle est utilisée lors de la linéarisation comme les autres mots de la phrase.

Nous plaçons également autour de chaque mot et ponctuation une balise, afin de pouvoir gérer par la suite finement la compression. Connaître ces délimitations nous permettra, au niveau de l'interface Web, par exemple :

- de compter les mots supprimés, pour les statistiques ;
- de gérer l'effacement des ponctuations en fonction de celui des constituants ;
- de contrôler les articles élidés[74].

Nous décrivons maintenant les principales grammaires de ce réseau.

Délimitation des constituants. Cette grammaire se charge d'ajouter deux variables à chaque nœud interne de l'arbre, une (DEBCONST) pour spécifier où commence le constituant dans la phrase d'entrée (en termes de nombre de caractères), l'autre (FINCONST) où il finit. Ces variables seront utilisées pour définir une position aux feuilles contenant les balises XML, afin qu'elles soient correctement placées après la linéarisation. En effet, la structure produite par SYGFRAN respectant assez bien la structure profonde de la phrase, certaines inversions de mots ou constituants peuvent survenir lors de la restitution de l'ordre original des éléments de la phrase, ce qui pourrait causer des mauvais encadrements de balises, et donc des suppressions de constituants erronées.

Les règles de cette grammaire permettent un parcours des arbres syntagmatiques des phrases, à partir des feuilles, en remontant de nœuds en nœuds, tout en propageant la position des mots dans la phrase (à partir de la variable PLACEMOT de SYGFRAN) au niveau des deux variables définies, pour chaque nœud interne.

[74] Par exemple, dans le groupe nominal *le gros animal*, si le modifieur *gros* est supprimé, alors l'article s'élide en *l'*.

Sachant que plusieurs constituants peuvent commencer ou finir sur un même mot, il est important de définir aussi une valeur d'inclusion des constituants, afin de correctement positionner les balises lors du tri des feuilles. Ainsi deux variables supplémentaires, `DECDEBCONST` et `DECFINCONST`, sont utilisées pour gérer l'inclusion des balises, sous la forme d'une valeur de décalage. Un constituant englobant un autre aura une valeur de début de décalage plus petite et une de fin plus grande. Lorsque les feuilles de balises seront ajoutées, elles hériteront des valeurs de décalage de début et de fin du constituant qu'elles encadrent, puis lors du tri des feuilles, ces valeurs seront ajoutées à celles de délimitation du constituant pour déterminer correctement l'ordre des feuilles.

L'arbre de la figure 4.16, basé sur la phrase de l'exemple 4.5, illustre l'utilisation de ces quatre variables.

Exemple 4.5 *Félix monte sur un toit glissant.*

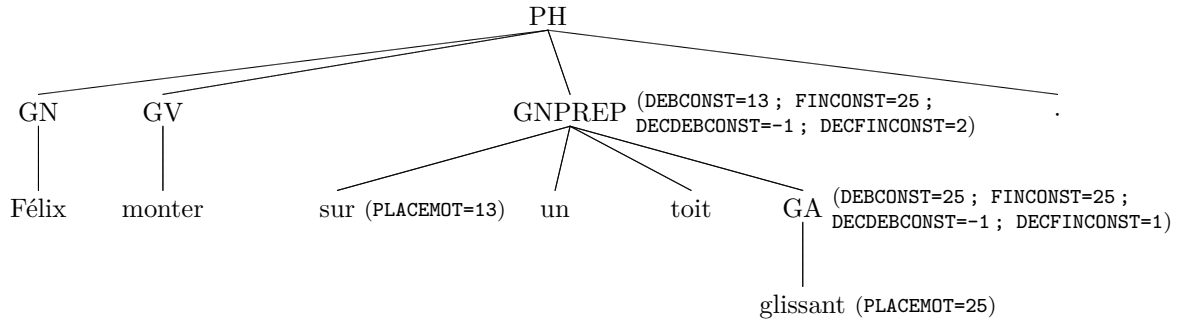

FIG. 4.16 – Exemple d'arbre avant linéarisation.

Le `GA` ne possédant qu'un seul mot se voit attribuer la valeur de `PLACEMOT` de son fils pour ses variables `DEBCONST` et `FINCONST`. Ses variables `DECDEBCONST` et `DECFINCONST` sont initialisées à -1 et 1 respectivement. Le `GNPREP` récupère la variable `PLACEMOT` de son fils tout à gauche pour renseigner sa variable `DEBCONST`. Son fils tout à droite ne possède pas de variable `PLACEMOT`, car il n'est pas un mot feuille, le `GNPREP` hérite alors de la variable `FINCONST` de ce fils. Enfin sa variable `DECDEBCONST` est initialisée à -1 et `DECFINCONST` récupère la valeur du fils incrémentée de 1. Nous verrons lors du tri des nœuds comment ces variables interviennent dans l'exemple précédent.

Encadrement des constituants. Nous plaçons maintenant les balises autour des mots et constituants. Pour chaque feuille de l'arbre de type mot, deux fils de type balise de mot (attribut `name` qui prend pour valeur `mot`) lui sont ajoutés, le fils gauche avec une balise ouvrante, l'autre une fermante. Pour chaque nœud de l'arbre qui voit sa variable `SUPPRTYPE` renseignée, c'est-à-dire le constituant est potentiellement supprimable, deux nœuds de

type balise de constituant (attribut `name` qui prend pour valeur `constituant`) sont ajoutés, en frères gauche et droit de ce nœud. La balise du frère gauche, celle qui ouvre, possède un attribut XML nommé `type` qui contient la valeur de la variable `SUPPRTYPE` utilisée dans COLIN, c'est-à-dire le type de constituant supprimable. Leur variable `PLACEMOT` est calculée à partir des variables de leur père : `DEBCONST` pour la balise ouvrante, et `FINCONST` pour la balise fermante. La feuille de balise ouvrante hérite de la variable `DECDEBCONST` de son père et celle de balise fermante hérite de la variable `DECFINCONST`. Ces deux variables seront utilisées lors du tri pour ordonner les balises entre elles.

Nous continuons notre exemple avec l'arbre de la figure 4.17 qui contient maintenant ces balises.

Les feuilles M_i sont des balises de mot, les P_i de ponctuation, et les C_i de constituant. Parmi ces nœuds, ceux d'indices impairs contiennent les balises XML ouvrantes, et les autres les fermantes.

Ainsi, les M_{2i+1} ont pour valeur de la variable `FRM` : `<span name="mot">`, pour $i \in [0, 5]$. Les M_{2i+2} contiennent la balise fermante : `</span>`. Les nœuds P_1 et P_2 ont pour valeur de la variable `FRM` : `<span name="ponctuation">` et `</span>`, respectivement.

L'arbre étant suffisamment chargé ainsi, nous renseignons les valeurs de variables pertinentes à cette partie dans le tableau 4.5.

	PLACEMOT	DECDEBCONST	DECFINCONST	FRM
C_1	13	-1	0	`<div name="constituant" type="MPLIEU">`
C_2	25	0	2	`</div>`
C_3	25	-1	0	`<div name="constituant" type="MN">`
C_4	25	0	1	`</div>`

TAB. 4.5 – Valeurs des variables des balises de constituant de l'arbre de la figure 4.17.

Avec `MPLIEU` pour modifieur de la proposition circonstant de lieu et `MN` pour modifieur du nom. Ce sont des balises `div` qui sont utilisées pour les constituants, car l'interface Web utilise un encadrement d'éléments HTML pour sélectionner les constituants, lequel nécessite une telle balise.

Aplatissement de l'arbre. L'arbre balisé est maintenant traité par une grammaire récursive qui :

- aplatit l'arbre en faisant remonter les feuilles et supprimant les nœuds internes, jusqu'à obtenir, pour chaque phrase, un arbre de hauteur deux, avec une racine et l'ensemble des mots et balises comme feuilles ;
- ordonne les feuilles de chaque phrase en se basant sur la variable `PLACEMOT`, puis, lorsque deux de ces variables sont identiques, les variables `DECDEBCONST` et `DECFINCONST` sont ajoutées à la première pour faire la différence.

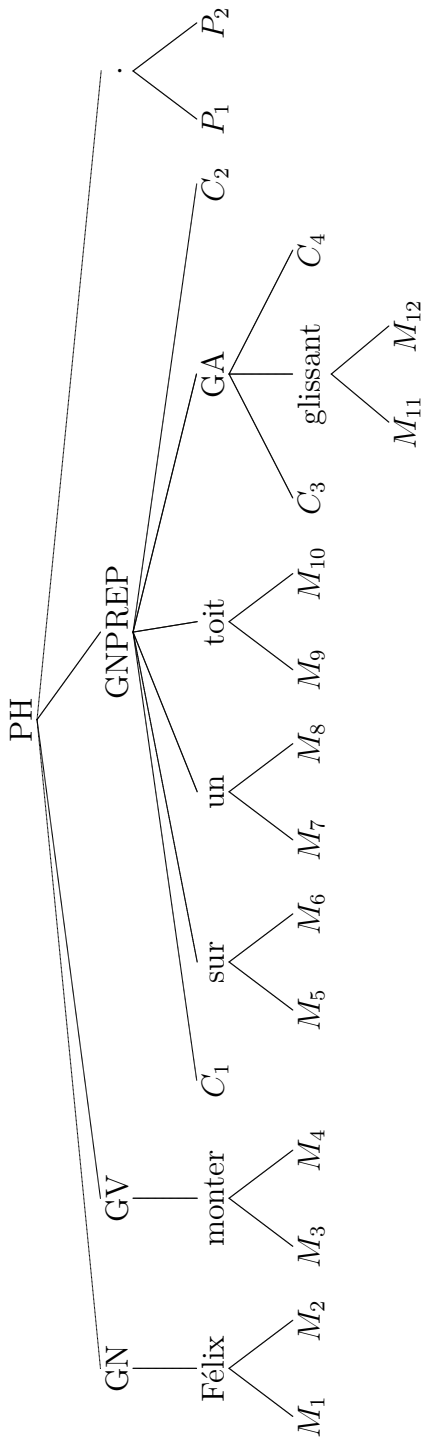

FIG. 4.17 – Exemple d'arbre balisé avant linéarisation.

L'arbre de la figure 4.18 représente l'arbre précédent aplati selon cette grammaire.

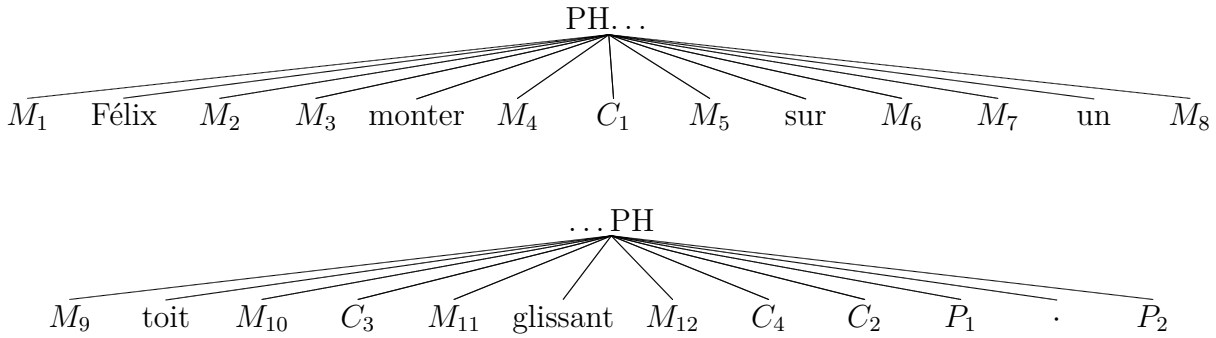

FIG. 4.18 – Exemple d'arbre aplati, avant linéarisation, avec M pour mot, C pour constituant et P pour ponctuation.

L'arbre a été découpé en deux, car trop large pour être présenté en entier sur une page.

Dans ce réseau de grammaires sont aussi traités :

- les variables de verrou de COLIN, renseignées par un attribut XML `verrou`, qui sera utilisé par l'interface Web pour pondérer l'importance des constituants concernés ;
- les retours à la ligne pour les alinéas et changements de paragraphes, afin de rétablir correctement la mise en page du texte compressé.

Nous ne détaillons pas ces grammaires, se référer au code source en annexe pour davantage de détails.

4.4.1.5 Grammaire de linéarisation

Enfin, l'arbre aplati est transmis au module AGATE de COLIN où les règles effectuent un parcours en profondeur des feuilles pour en extraire les valeurs des variables `FRM` et les concaténer pour produire la chaîne de sortie, c'est-à-dire notre texte étiqueté, prêt à être manipulé par l'interface Web pour réaliser les différentes compressions possibles.

Par exemple, l'arbre aplati de la figure 4.18 permettra de générer la chaîne (simplifiée et indentée) de la figure 4.19.

4.4.2 Interface Web de COLIN

Comme nous l'avons vu en début de la section 4.4, COLIN comprend une interface Web permettant à l'utilisateur de compresser un texte, à travers une interaction. Le procédé d'interaction dans le résumé automatique est rare. Nous commençons cette section par présenter l'approche du domaine que nous avons jugée la plus significative sur ce point, section 4.4.2.1, puis nous présentons la nôtre, à travers notre interface Web, section 4.4.2.2.

```
<span name="mot">Félix</span>
<span name="mot">monte</span>
<div name="constituant" type="MPLIEU">
        <span name="mot">sur</span>
        <span name="mot">un</span>
        <span name="mot">toit</span>
        <div name="constituant" type="MN">
                <span name="mot">glissant</span>
        </div>
</div>
<span name="ponctuation">.</span>
```

FIG. 4.19 – Exemple de chaîne de sortie du module OPALE de COLIN.

4.4.2.1 L'interaction dans le résumé automatique

Une interaction ralentit considérablement le traitement d'une donnée, mais permet en contrepartie d'en augmenter la qualité. Dans [Leuski *et al.*, 2003], les auteurs mettent en avant l'intérêt d'une telle intervention de l'utilisateur (le récepteur) dans leur résumeur multi-document : « *The quality of the summary depends strongly on users' present need — a summary that focuses on one of several topics contained in the material may prove to be either very useful or completely useless depending on what users' interests are.* » (La qualité du résumé dépend fortement du besoin actuel de l'utilisateur — un résumé se concentrant sur un des thèmes contenus dans le document source peut s'avérer utile ou complètement inutile selon quels sont les intérêts de l'utilisateur.)

Dans cette approche, les auteurs ont fait le choix de proposer à l'utilisateur un nombre conséquent de paramètres, lui offrant ainsi une grande souplesse de modelage du résumé, au détriment d'une complexité accrue de maîtrise du système. Voici les différents paramétrages possibles :

- changer la taille du résumé ;
- définir la position limite dans chaque document des phrases à conserver ;
- modifier la valeur de chevauchement des phrases entre les différents documents (la redondance) ;
- modifier les signatures des différents thèmes des documents identifiés par le système, par ajout ou suppression de lexies ;
- ajouter des nouveaux thèmes en leur définissant des signatures propres ;
- sélectionner/dé-sélectionner des sous-ensembles de thèmes ;
- naviguer parmi les documents à résumer grâce à un système d'aperçu et de lien entre les phrases du résumer et celles des documents sources.

Ainsi l'utilisateur va paramétrer le système, en observant progressivement les changements

sur le résumé final, jusqu'à ce qu'il obtienne un résumé maximisant sa satisfaction. Les auteurs n'ont pas effectué d'évaluation de leur système, il est donc difficile de connaître le réel apport d'une telle approche. Ils défendent l'intérêt de l'interaction de l'utilisateur avec le système semi-automatique en citant une expérimentation en recherche d'information, décrite dans [Koenemann & Belkin, 1996], dans laquelle est établi un lien notable entre le niveau d'interaction et la qualité des résultats obtenus.

Cette approche permet donc à l'utilisateur de configurer le système défini par les concepteurs. Les limites de contrôle de l'utilisateur sont définies par les possibilités de paramétrage qui lui sont accordées, par la méthode de résumé utilisée, mais aussi par l'impossibilité de manipuler directement le résumé final. L'augmentation du nombre de paramétrages possibles augmente à la fois les chances de se rapprocher d'un résumé satisfaisant mais aussi les complexités d'apprentissage et d'utilisation du système.

Afin d'éviter un paramétrage complexe et une modification indirecte du résumé produit, nous avons opté, dans COLIN, pour une interaction oblique à celle-ci, que nous décrivons dans la prochaine section.

4.4.2.2 L'interaction dans COLIN

Nos règles de compression associent à chaque constituant potentiellement effaçable une variable de type de suppression, créant alors des classes de constituants supprimables. Dans le texte étiqueté traité par l'interface, ces constituants sont délimités par des balises XML, et leur type de suppression est défini par leur attribut `type`.

Considérant cette donnée, notre première interaction s'orienta vers un paramétrage global, où l'utilisateur pouvait activer ou désactiver la suppression de chaque classe de constituant. Le paramétrage global ne satisfaisant plus nos exigences en termes de qualité de résumé produit, nous avons par la suite opté pour une interaction où le système donne les moyens à l'utilisateur d'intervenir sur le choix de suppression de chaque unité textuelle, c'est-à-dire le constituant dans notre cas, plutôt que sur un paramétrage global.

Le but de cette nouvelle interface est alors de mettre en valeur les constituants potentiellement effaçables, d'après l'étiquetage du texte généré par les règles transformationnelles de COLIN, et de permettre une interaction directe sur ces constituants, en confirmant ou en infirmant leur sélection.

Si l'utilisateur fait confiance à la sélection des constituants faite par le compresseur, alors il peut valider directement le résumé. On est alors dans un cas de résumé sans réelle interaction, donc automatique. Si l'utilisateur intervient sur la sélection, le résumé devient alors semi-automatique, ce qui est le sujet de cette section.

Cette méthode d'interaction a l'avantage, sur celle décrite dans la section précédente, de permettre un résumé de meilleure qualité, car la sélection du contenu important est

directe et locale, plutôt qu'indirecte et globale. L'inconvénient est naturellement le temps de résumé qui s'accroît, car l'utilisateur va très certainement parcourir l'ensemble du texte à résumer avant de le valider, contrairement au résumé par paramétrage global, où l'interaction devrait être plus courte.

Nous décrivons maintenant comment se déroule l'interaction locale de CoLin, puis nous présentons le fonctionnement technique de l'interface.

La sélection des constituants. Après avoir récupéré le texte étiqueté, notre interface Web l'affiche dans son intégralité, en encadrant chaque constituant sélectionnable, c'est-à-dire potentiellement effaçable (modifieurs et compléments, sauf quelques exceptions). Un constituant encadré est un élément interactif sur lequel l'utilisateur peut cliquer pour le sélectionner ou dé-sélectionner, et ainsi le supprimer ou le conserver dans la compression. Lorsque le constituant est sélectionné, le cadre est plein, sinon il est vide. Un cadre plein est visuellement similaire à un surlignage, ne masquant pas le constituant sélectionné. Ainsi l'utilisateur peut, à tout moment, visualiser quelle sera la compression finale, tout en gardant en vue les éléments qui seront supprimés.

Des couleurs pour l'importance. Tous les constituants sélectionnables ne disposent cependant pas de la même importance, comme nous l'avons vu dans le précédent chapitre. L'importance relative de chaque classe de constituants n'est toutefois pas toujours facilement estimable. C'est pourquoi, à partir de notre étude théorique ainsi que nos premières expériences de compression, nous avons regroupé ces classes en trois catégories d'importance **probable** : importance généralement faible, importance variable, importance généralement élevée.

Puis nous avons associé à chacune de ces catégories une couleur, afin d'indiquer visuellement à l'utilisateur l'importance des constituants estimée par le système. Les couleurs sont le vert, pour une importance faible, le jaune, pour variable et le rouge, pour élevée. Ces couleurs sont appliquées au cadre des constituants, ainsi qu'à leur surlignage. Enfin, ces catégories déterminent la sélection par défaut des constituants : les constituants généralement importants (rouges) ne seront pas sélectionnés par défaut, les autres le seront.

Ainsi, la sélection par défaut respecte notre étude sur l'importance des constituants, car ceux qui sont non effaçables ne sont pas sélectionnables, et ceux qui sont sélectionnables mais généralement importants ne sont pas sélectionnés par défaut. Si l'utilisateur valide la sélection par défaut, il obtient donc une compression qui devrait vérifier au mieux les contraintes de conservation du contenu important, selon notre étude théorique et les informations disponibles dans l'analyse syntaxique.

Si l'utilisateur préfère plutôt contrôler plus finement la compression, il peut ainsi modifier la sélection, constituant par constituant. Il peut alors s'appuyer sur la couleur pour

optimiser la répartition de son temps d'analyse, selon le taux et la qualité de compression qu'il souhaite obtenir. Par exemple, s'il souhaite une compression de qualité élevée, il pourra ignorer les constituants encadrés de rouge et se concentrer sur les jaunes et verts. S'il souhaite par contre un taux de compression élevé, il pourra vérifier chaque constituant encadré en rouge pour voir si certains ne sont pas finalement supprimables sans trop de perte de contenu important.

L'inclusion des constituants. Il arrive très souvent que des constituants sélectionnables s'imbriquent (lorsqu'ils partagent des mêmes mots). Dans ce cas, les cadres de sélections s'imbriquent aussi et puisqu'un mot ne peut être colorié de plusieurs couleurs à la fois, il faut définir une politique de coloration des constituants imbriqués. Afin de limiter la surcharge de couleurs, nous avons décidé de rendre prioritaire le surlignage du constituant le plus incluant, pourvu qu'il soit sélectionné. Pour des raisons de clarté visuelle, tous les cadres des constituants inclus dans celui qui est sélectionné ne sont pas non plus visibles. La dé-sélection de ce constituant fera alors apparaître les cadres de ses fils par inclusion.

Outre la coloration, il est aussi nécessaire de définir une politique d'interaction sur les constituants imbriqués. En effet, si l'utilisateur clique sur un mot appartenant à plusieurs constituants, alors lequel ou lesquels doivent être (dé)sélectionnés ? Après des tests d'ergonomie de l'interface, nous avons décidé que :

- si le clic est effectué sur un surlignage, alors c'est le constituant sélectionné le plus englobant qui est prioritaire sur l'interaction ;
- si le clic est effectué sur un cadre vide, alors c'est le constituant le plus englobé qui est prioritaire sur l'interaction.

De plus, lorsqu'un constituant est sélectionné, tous les constituants qu'il englobe le sont aussi, afin qu'une tête de constituant ne puisse être sélectionnée sans que ses constituants gouvernés ne le soient aussi, ce qui causerait une incohérence grammaticale. Enfin, lorsqu'un constituant est dé-sélectionné, ses fils d'inclusion restent sélectionnés, sauf s'ils sont rouges (probablement importants), et ceci récursivement.

Exemple de capture d'écran de l'interface de CoLin. La figure 4.20 est une telle capture d'écran, elle illustre la sélection par défaut des constituants, ainsi que l'inclusion de certains constituants.

Dans cet exemple, un court texte de quatre phrases, du genre narratif, a été analysé. Si l'utilisateur clique sur le bouton « Générer le texte compressé », alors le texte suivant est généré en dessous du bouton :

« Un chat vit deux souris. Le prédateur décida de les attraper. Il s'approcha puis bondit. Elles furent effrayées par l'animal affamé. »

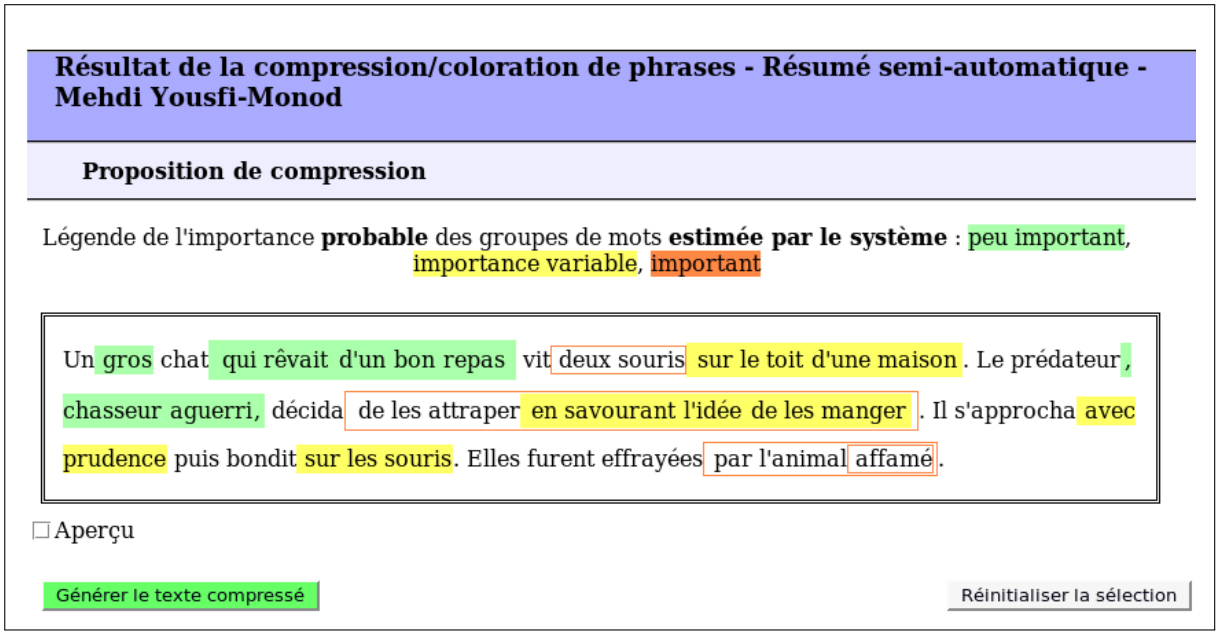

FIG. 4.20 – Capture d'écran de l'interface de COLIN.

Cette sélection peut ne pas convenir à l'utilisateur, il peut par exemple juger que les constituants *sur le toit d'une maison* et *sur les souris* sont suffisamment importants pour être conservés, et que le constituant *par l'animal affamé* ne l'est pas. Il clique alors sur ces trois constituants, et obtient la sélection de la figure 4.21.

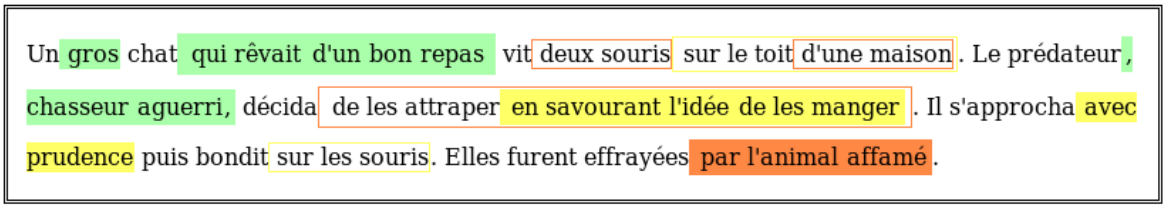

FIG. 4.21 – Exemple de modification de la sélection par défaut.

Nous pouvons observer ici le surlignage rouge qui attire l'attention, signalant à l'utilisateur qu'il a choisi d'effacer un constituant auquel le système attribuait une importance probable élevée. Nous remarquons aussi que la dé-sélection du constituant *sur le toit d'une maison* fait apparaître un constituant sélectionnable *d'une maison*, mais attribué d'une couleur rouge et donc dé-sélectionné par défaut. Ce constituant est un modifieur du nom précédé d'un article définit, lequel est généralement important.

La nouvelle compression obtenue après validation est alors :

« Un chat vit deux souris sur le toit d'une maison. Le prédateur décida de les attraper. Il s'approcha puis bondit sur les souris. Elles furent effrayées. »

Le fonctionnement technique de l'interface Web de COLIN. L'interface repose entièrement sur des technologies Web. Ces technologies étant suffisantes pour l'interaction que nous souhaitions, nous avons préféré faire ce choix plutôt que mettre en œuvre un client externe, nécessitant une installation particulière, une gestion du réseau supplémentaire et une programmation plus lourde. L'interface Web a ainsi les avantages de faciliter l'accès pour l'utilisateur et le développement pour le concepteur.

Elle est générée par un programme PHP interprété par le logiciel Apache HTTP Server d'un serveur du LIRMM[75]. Cette interface génère des codes source HTML, CSS et Javascript qui sont interprétés par le navigateur Web de l'utilisateur.

La partie interactive est gérée par Javascript. Voici les principales tâches effectuées par notre programme Javascript :

- attachement d'un événement sur le clic pour chaque constituant sélectionnable ;
- redéfinition du comportement des événements pour l'inclusion des éléments HTML :
  - passage en mode *bubble*, c'est-à-dire où l'élément le plus imbriqué touchant au clic génère en premier l'événement ;
  - stoppage de la propagation des événements, ainsi seul l'élément le plus imbriqué génère un événement[76].
- définition de la couleur et la taille des cadres, selon le niveau d'inclusion ainsi que l'interligne, selon la hauteur des cadres ;
- définition de la (dé)sélection lors d'un clic sur un constituant, ainsi que la propagation de la sélection aux constituants imbriqués et imbriquants, comme vu en section 4.4.2.2, cela est effectué grâce à la navigation dans l'arborescence du document HTML en utilisant la syntaxe DOM ;
- définition d'une fonction de compression qui parcourt les mots, conserve ceux qui ne sont pas dans un constituant sélectionné, puis les concatène pour la sortie[77] ;
- définition d'une fonction de comptage dynamique des mots sélectionnés, afin d'afficher le taux de compression du futur résumé en temps réel ;
- définition d'une fonction d'initialisation de la coloration, qui peut aussi être appelée par le bouton « Réinitialisation de la sélection » de l'interface ;
- définition d'une fonction d'aperçu du résultat qui permet, via un bouton radio HTML, d'effacer les constituants sélectionnés, en fixant à `hidden` la variable `visibility`, propriété CSS des balises de constituants.

[75] Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier, laboratoire dans lequel j'ai effectué ma thèse. Site Internet : `http://www.lirmm.fr`

[76] Ce comportement n'est pas géré correctement sur tous les navigateurs Web. Ceux basés sur le moteur de rendu Gecko, comme Mozilla Firefox ou Netscape, ceux basés sur le moteur KHTML, comme Konqueror ou Safari, ou encore d'autres comme Opera, fonctionnent correctement sur cet aspect. Par contre, Internet Explorer échoue à détecter correctement l'élément qui a généré l'événement.

[77] Cette fonction gère aussi la suppression éventuelle des ponctuations ainsi que les élisions d'articles.

4.5 Conclusion

Dans ce chapitre, nous avons présenté notre mise en œuvre du compresseur de phrases. Nous avons commencé par décrire, section 4.2, l'architecture globale d'un système de compression automatique vers laquelle nous souhaitions nous diriger. Cette dernière met en jeu deux principaux composants qui sont l'analyseur syntaxique et le compresseur de phrases. Les deux sections suivantes présentent alors l'analyseur syntaxique que nous avons choisi ainsi que le compresseur de phrases que nous avons développé.

L'analyseur syntaxique, présenté en section 4.3, est SYGFRAN. Il produit une analyse morpho-syntaxique des phrases du français, correspondant étroitement à notre grammaire définie dans le chapitre précédent. SYGFRAN s'appuie sur le système opérationnel SYGMART, lequel permet de définir des règles de décomposition morphologique d'une chaîne de caractères pour ensuite produire un arbre sur les mots identifiés, et enfin manipuler cette structure arborescente par des règles transformationnelles. SYGFRAN est alors un programme pour SYGMART, composé d'un ensemble de règles et de dictionnaires, qui produit un arbre syntagmatique enrichi d'informations linguistiques sur chaque feuille (mots) et nœud interne (constituant) de l'arbre produit.

Notre compresseur de phrases, COLIN, présenté en section 4.4, est composé d'un ensemble de règles pour SYGMART et d'une interface Web pour interagir avec un utilisateur. Les règles sont interprétées par SYGMART, et récupèrent directement la structure produite par SYGFRAN. Ces règles consistent à adapter la syntaxe de SYGFRAN à la nôtre, ainsi que sa terminologie, puis à marquer chaque constituant potentiellement effaçable d'une variable de type de suppression, pour ensuite linéariser la structure arborescente en y ajoutant des balises encadrant les constituants marqués et intégrant la variable de suppression, afin de pouvoir les manipuler au niveau de l'interface. Cette interface se charge de présenter à l'utilisateur la proposition de compression de COLIN. L'utilisateur peut alors soit accepter la proposition, c'est-à-dire la sélection par défaut des constituants, et ainsi valider directement la compression, soit adapter la proposition à ses préférences, en interagissant directement avec chaque constituant sélectionnable proposé par l'interface. Dans le premier cas, la compression sera rapide, mais circonscrite aux performances du compresseur et de l'analyseur syntaxique, dans le second cas, la compression dépendra du temps d'interaction, et proposera un résumé plus près des attentes de l'utilisateur. L'interface repose sur des technologies Web et est accessible directement depuis un navigateur Web.

Dans le prochain chapitre, nous présentons une évaluation de COLIN, en mode automatique et semi-automatique, sur un corpus de textes de trois genres différents.

5

Évaluation de notre approche sur la compression syntaxique des phrases

Sommaire

5.1 Introduction 129
5.2 L'évaluation de résumés 130
5.3 Protocole d'évaluation de CoLin 134
5.4 L'expérimentation 147
5.5 Conclusion 161

5.1 Introduction

L'étude de notre modèle de compression syntaxique computationnel ainsi que la présentation de notre compresseur automatique étant effectuées, nous abordons maintenant la dernière étape du paradigme scientifique classique, laquelle consiste à valider notre approche théorique au travers d'une expérimentation et de l'interprétation de ses résultats.

Notre système de validation utilise naturellement notre compresseur, CoLin, pour réaliser les compressions automatiques et semi-automatiques de l'expérimentation. La validation reste donc circonscrite aux éléments théoriques de notre étude mis en œuvre dans le compresseur, ainsi qu'aux limites de l'analyseur syntaxique, qui fournit les données indispensables au bon fonctionnement du compresseur. Cependant, comme nous en avons discuté dans le précédent chapitre, la limite de la qualité d'analyse peut être repoussée dans le cadre du traitement d'un corpus de documents limité, par un traitement *ad hoc* de la donnée en entrée. Ainsi, notre compresseur disposant d'une information syntaxique de bonne qualité, cette évaluation peut constituer une validation assez pertinente de notre approche.

Nous avons emprunté un chemin différent de la majorité des approches au sujet du protocole d'évaluation. Nous présentons la tendance classique dans le domaine, section 5.2, puis nous définissons notre protocole et notre système d'évaluation, section 5.3. Nous terminons le chapitre par la description de l'expérimentation ainsi que l'étude et la discussion sur ses résultats, section 5.4.

5.2 L'évaluation de résumés

Le but de la production d'un résumé est de proposer d'une part un texte plus court que l'original et d'autre part un texte qui représente bien l'original. Plus le résumé est court, plus il est difficile de bien représenter l'original. L'évaluation de la qualité d'un résumé doit alors prendre en compte ces deux facteurs. Ainsi, pour un texte T à résumer, un résumé R_1 sera meilleur qu'un résumé R_2 si, à tailles égales, R_1 représente mieux T que R_2, ou à niveaux de représentations égaux, R_1 est plus petit que R_2.

La taille d'un résumé est facilement calculable, en nombre de mots ou de caractères par exemple, cependant, estimer son niveau de représentation relève d'une tâche bien plus difficile. Les principales raisons sont que d'une part l'appréciation de ce critère par un humain (évaluation manuelle) est subjective et donc variable selon les individus, et d'autre part son appréciation par un système automatique (évaluation automatique) n'est que partiellement réalisable avec les techniques actuelles. Le problème de la subjectivité peut amener à des situations où un même résumé est évalué très différemment par plusieurs humains différents. Le problème du traitement automatique est que l'évaluation peut être biaisée, ce qui peut amener à surestimer ou sous-estimer la qualité d'un résumé.

Essayer ainsi d'estimer directement la qualité d'un résumé par rapport au texte original est appelé une évaluation intrinsèque. Une autre façon de procéder pour évaluer la qualité d'un résumé est de mesurer à quel point ce résumé est utile dans l'accomplissement d'une tâche donnée. Par exemple, pour un document qui contient les réponses à en ensemble de questions, si son résumé suffit aux lecteurs pour répondre aux questions, alors il constitue un bon représentant. Nous avons fait le choix, pour notre évaluation, d'utiliser une évaluation intrinsèque car nous avons jugé sa mise en œuvre plus abordable dans le cadre de notre travail.

Nous présentons maintenant quelques approches qui traitent d'évaluation intrinsèque, automatiques ou manuelles, ainsi que les choix qui nous on conduits vers notre propre évaluation.

5.2.1 Évaluation automatique

L'évaluation manuelle des systèmes de traitement automatique du langage naturel est une tâche coûteuse en temps et en efforts. En effet, elle nécessite un travail fastidieux d'un ensemble d'évaluateurs, et ceci pour chaque résumé à évaluer. Afin de trouver une alternative rapide et moins chère pour l'évaluation, des méthodes automatiques et statistiques ont été proposées depuis quelques années.

[Donaway *et al.*, 2000] proposent une étude comparative entre trois méthodes automatiques d'évaluation de résumé par extraction de phrases.

La première, basée sur un calcul de rappel, procède par comparaison des résumés automatiques avec une sélection de phrases importantes faite par des juges humains. Le principal problème soulevé par les auteurs est que la mesure de type rappel introduit un biais, car elle est basée sur l'opinion d'un petit groupe d'évaluateurs (problème de subjectivité précédemment abordé). À ce sujet, [Jing *et al.*, 1998] corroborent le défaut de ce genre de technique par ces mots : « *... precision and recall are not the best measures for computing document quality. This is due to the fact that a small change in the summary output [...] can dramatically affect a system's score.* » (la précision et le rappel ne sont pas les meilleures mesures pour calculer la qualité d'un document. Cela est dû au fait qu'un petit changement dans le résumé produit [...] peut affecter considérablement le score du système). [Saggion & Lapalme, 2002] utilisent une méthode similaire, appelée « Évaluation du contenu dans une expérience de cosélection », qui est naturellement affublée du même défaut.

La seconde méthode consiste à demander aux juges de trier les phrases de chaque texte à résumer par ordre d'importance. L'ordre moyen d'importance des phrases obtenu est alors utilisé pour être comparé aux résumés automatiques, par un calcul encore une fois basé sur le rappel. Bien que les auteurs mettent avant quelques avantages de cette méthode sur la précédente, elle reste circonscrite aux problèmes de calculs basés sur le rappel.

Enfin, la troisième méthode se concentre sur la conservation du contenu informationnel. Le but est de comparer le contenu du résumé automatique avec soit le texte original, soit un résumé humain. L'évaluation s'effectue par une comparaison de calculs de fréquence de termes, basés sur le quotien $tf \times idf$, et souple à l'égard des synonymes, grâce à l'utilisation d'un thésaurus. Ainsi ce genre d'approche évalue la qualité d'un résumé à la quantité de vocabulaire partagé avec son texte original ou un autre résumé produit par un humain. Bien que cette méthode puisse révéler une certaine efficacité pour le résumé par extraction de phrases, elle s'avère peu efficace lorsqu'il s'agit d'évaluer un résumé par compression de phrases. Le principal problème est que la grammaticalité des phrases n'est pas du tout

évaluée. Par exemple, si un système de résumé compresse les phrases en ne conservant que les mots clés, les résumés produits (des listes de mots), pourront obtenir un bon score avec une telle méthode. Un autre problème est que la modification du sens des mots, affectée par une modification de la structure syntaxique, n'est pas évaluée. Par exemple, la suppression d'un complément peut profondément modifier le sens de sa tête.

5.2.1.1 ROUGE

ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) [Lin *et al.*, 2003] est une méthode d'évaluation automatique de résumé qui est utilisée dans la compétition de DUC[78] (*Document Understanding Conferences*), depuis l'édition 2004, comme seule mesure de fiabilité pour certaines tâches. Cette méthode fait référence, à l'heure actuelle, dans le domaine du résumé automatique.

ROUGE, basée sur le calcul statistique de co-occurrence de n-grammes de mots, est adaptée de BLEU [Papineni *et al.*, 2002], utilisée pour la traduction automatique. Ces n-grammes sont des groupes de n mots consécutifs. ROUGE évalue la qualité d'un résumé automatique par le nombre de n-grammes qu'il partage avec un résumé de référence. Plus ce nombre est important et meilleure est la qualité.

D'après les auteurs de [Lin *et al.*, 2003], l'unité la plus pertinente pour l'évaluation de ROUGE, est l'uni-gramme (lorsque n vaut 1), avec laquelle la méthode offre des résultats qui sont assez bien corrélés avec les jugements humains, en précision et rappel. Lorsqu'une valeur de n supérieure à 1 est considérée, l'efficacité de ROUGE chute rapidement. Les créateurs de cette métrique pensent que les n-grammes plus longs évaluent davantage la grammaticalité des phrases que le choix du contenu important.

Or, la majorité des approches évaluées dans DUC se soucient peu de la grammaticalité, car leur unité textuelle de résumé est la phrase, laquelle est supprimée ou conservée dans son intégralité dans le résumé produit. Utiliser des uni-grammes ne permet pas d'évaluer la grammaticalité des phrases, alors qu'utiliser des n-grammes plus longs est peu pertinent vis-à-vis de la conservation du contenu. Notre approche de compression de phrases, visant à conserver à la fois le contenu important et la grammaticalité des phrases, ne semble donc pas adaptée à l'évaluation basée sur des n-grammes.

De plus, même avec des uni-grammes, la corrélation avec les résumés humains n'est pas toujours satisfaisante. Les auteurs le précisent en conclusion : « *Although this study shows that unigram co-occurrence statistics exhibit some good properties in summary evaluation, it still does not correlate to human assessment 100 % of the time* » (Bien que cette étude révèle que les statistiques de co-occurence d'unigrammes expose de bonnes propriétés pour l'évaluation de résumés, cela ne corrèle tout de même pas à 100 % avec les jugements

[78] Site internet : `http://duc.nist.gov/`

humains).

Enfin, comme nous l'avons vu en début de section, deux bons résumés peuvent utiliser des mots différents, conserver des phrases différentes, tout en étant de bons représentants du document original, mais considérant des critères d'importance différents, subjectifs à l'auteur du résumé. De ce fait, se baser sur les n-grammes pour estimer la qualité d'un résumé, peut produire des résultats différents selon la nature du résumé de référence considéré. L'évaluation basée sur des résumés de référence est donc limitée par cette contrainte inhérente à ce procédé. Afin de réduire cet effet négatif, l'évaluation dans DUC s'est appuyée sur plusieurs résumés de référence, pour chaque document et chaque tâche. Cependant le nombre de tels résumés de référence est naturellement limité, et leur exploitation simultanée pour estimer la qualité d'un résumé automatique ne peut être parfaite.

Pour ces raisons, nous avons préféré traiter l'évaluation de notre approche par une technique manuelle.

5.2.2 Évaluation manuelle

L'approche de Knight et Marcu, désormais K&M, décrite en section 2.3.3.5, utilise un protocole d'évaluation plus adapté à la compression de phrases. Leur modèle nécessite un apprentissage sur un corpus de phrases associées (sous forme de paire) à leur compression réalisée par un humain. Pour l'évaluation, ils ont utilisé une partie du corpus pour l'apprentissage, 1035 paires de phrases, et une autre pour l'évaluation, 32 paires de phrases tirées au hasard dans le corpus. Le genre de leur corpus était des articles journalistiques d'annonces de produits informatiques.

Ils ont aussi réalisé une évaluation à partir d'un petit corpus scientifique (26 phrases), afin de tester l'influence du genre sur leurs algorithmes d'apprentissage.

Dans leurs deux évaluations, quatre types de compression ont été exploités :

- celle issue de leur modèle probabiliste ;
- celle issue de leur modèle basé sur la décision ;
- celle générée par l'humain ;
- une issue d'un algorithme de base[79], utilisée comme niveau de référence (*baseline*).

Pour chaque phrase originale, chaque version compressée fut présentée dans un ordre aléatoire à 4 juges humains. Il fut dit aux juges que les quatre versions avaient été générées de manière automatique. Puis l'évaluation se déroula en deux étapes. Dans la première, les juges devaient donner une note entre 1 et 5 sur la conservation des mots importants

[79] Les phrases du corpus scientifique étant parfois très longues, cet algorithme, basé sur un calcul de scores de bi-grammes, n'a cependant pas pu fournir certaines phrases de ce type de compression, pour des raisons de temps de calcul trop longs.

dans la phrase résumée, puis dans la seconde, une note entre 1 et 5 sur le niveau de grammaticalité de la phrase.

Leur méthode d'évaluation a l'avantage sur ROUGE de prendre en compte la granularité plus fine des unités textuelles analysées dans le cadre d'une compression de phrases, grâce au critère de grammaticalité. Elle nécessite cependant deux passes, chacune imposant aux évaluateurs de se concentrer sur des tâches peu naturelles et éprouvantes cognitivement :

- déterminer les mots importants dans les phrases source et les comptabiliser dans les phrases cibles, sans tenir compte de la grammaticalité des phrases ;
- puis ne considérer que la cohérence syntaxique des phrases, sans se soucier du contenu.

De plus, juger de la qualité de phrases résumées, hors de tout contexte, n'est pas une tâche habituelle pour un humain, qui considérera plus facilement la qualité d'un ensemble de phrases formant un texte cohérent et complet. Travailler sur des phrases isolées accroît alors la charge cognitive de l'évaluateur.

Le travail de l'évaluateur étant assez difficile, il est alors peu évident d'en réunir un grand nombre dans le cadre d'une évaluation bénévole.

Toutefois, la méthode nous satisfait dans les grandes lignes, nous nous en sommes alors appuyés pour définir notre protocole d'évaluation, que nous décrivons dans la prochaine section. Nous n'avons cependant pas pu nous comparer à cette approche sur le même corpus, car notre analyseur syntaxique s'applique au français, alors que le corpus utilisé dans l'approche décrite ici est en anglais.

5.3 Protocole d'évaluation de CoLin

Nous définissons maintenant le protocole qui nous a permis d'évaluer CoLin. Ainsi nous utilisons une évaluation manuelle, plutôt que statistique, afin d'obtenir une estimation de la qualité des compressions qui n'est pas dépendante d'un résumé de référence et d'un modèle statistique. Notre évaluation implique un ensemble d'humains qui auront à compresser et/ou noter un ensemble de données textuelles. Nous appelons *évaluateurs* ces humains participant à notre évaluation pour ces deux tâches possibles.

CoLin disposant d'un mode semi-automatique (interactif), d'aide au résumé automatique, nous évaluons tout d'abord la qualité de cette aide apportée à l'utilisateur, à travers un protocole que nous présentons section 5.3.1. Nous abordons ensuite l'évaluation de la qualité des compressions produites par les versions semi-automatique et automatique de notre compresseur. Nous nous appuyons sur le protocole de K&M pour définir le nôtre, section 5.3.2.

5.3.1 Protocole d'évaluation de l'aide apportée par l'interaction dans COLIN

Dans son mode interactif, COLIN vise à aider un humain à produire un résumé par compression de phrases. Nous évaluons l'aide apportée par comparaison avec une tâche de compression similaire mais manuelle.

Ainsi, certains évaluateurs devrons réaliser des compressions manuelles, en respectant le mode de résumé par compression de phrases, alors que d'autres utiliseront l'outil interactif pour cette même tâche. Nous proposons à ces deux catégories d'évaluateurs des textes complets plutôt que, comme le font K&M, des phrases indépendantes afin de leur faciliter l'interprétation de l'énoncé et donc l'estimation des parties importantes. Nous contraignons la production du résumé manuel au mode de compression de phrases afin que les évaluateurs ne s'orientent pas vers une production de résumés plus naturelle pour un humain, c'est-à-dire mettant en jeu des procédés de reformulation, mais aussi des suppressions de phrases complètes. Si cela se produisait, les résumés produits manuellement ne seraient alors pas comparables aux compressions de phrases produites par les modes automatique et semi-automatique et l'évaluation ne serait pas pertinente. Cette contrainte de mode de production est identique à celle imposée aux experts qui ont constitué le corpus d'apprentissage utilisé par K&M.

L'aide apportée par COLIN peut être analysée sur au moins 4 critères :

1. le temps gagné ;
2. l'effort cognitif allégé ;
3. la satisfaction d'utilisation de l'outil ;
4. la qualité des compressions produites.

Pour évaluer le premier critère, il suffit de compter le temps pris par chaque compression, semi-automatique comme manuelle, puis de comparer les résultats. Afin que ce critère ne soit pas biaisé, nous avons prévenu les évaluateurs que leur temps de compression serait compté. Les secondes passées sont affichées durant la tâche et il est possible à tout moment de faire une pause, pour les situations imprévues où l'évaluateur doit cesser momentanément l'évaluation. Durant la pause, le texte à compresser est masqué, afin que l'évaluateur n'en profite pas pour réfléchir sur la compression.

Le second critère est plus difficile à analyser. D'une part, la tâche demandée n'est pas naturelle pour l'humain, qui procède habituellement différemment pour produire des résumés. D'autre part, il y a peu d'indices qui peuvent révéler un effort cognitif plus intense. Lorsqu'une tâche est plus éprouvante sur ce point, l'humain aura tendance à prendre plus de temps pour la réaliser, et en général plus de temps de récupération entre deux compressions. Cependant, le temps de résumé est déjà comptabilisé dans le premier

critère et le temps de récupération peut être facilement biaisé, car l'évaluateur peut vaquer à d'autres occupations entre deux compressions, et il est difficile de lui demander de toutes les enchaîner d'un trait. Pour ces raisons, nous considérerons l'influence de ce critère qu'à travers le premier.

Le troisième critère mesure l'impression de satisfaction globale de l'évaluateur, suite à l'interaction avec l'interface de COLIN. Cette donnée nous permet d'évaluer l'union d'un ensemble de qualités de l'interface, comme l'ergonomie et le plaisir de l'interaction. Une satisfaction élevée implique une interaction plus facile et motivante, et ainsi une aide au résumé plus élevée.

Enfin, le quatrième critère sera évalué grâce à la notation des résumés, décrite dans la prochaine section.

5.3.2 Protocole d'évaluation de la qualité des compressions produites par COLIN

Après nous être intéressés à l'évaluation de l'interaction de notre compresseur, nous nous intéressons maintenant à l'évaluation de la qualité des compressions produites par COLIN, en mode semi-automatique comme automatique.

Notre premier protocole d'évaluation s'appliquait à l'ancienne version du compresseur, appelée prototype. Nous avions défini un mode automatique de l'outil en ordonnant les classes de constituants sélectionnables par ordre d'importance probable, puis un fixant un seuil d'importance pour ne supprimer que les constituants appartenant aux classes en dessous du seuil. L'évaluation consista à comparer, pour un ensemble de textes, les choix de sélection des mots (à supprimer) du prototype avec ceux d'un ensemble d'experts. Pour cela, nous avions choisi un calcul à base de F-Score comme métrique d'évaluation. Le résultat fut faible, avec environ 0,2 de F-Score moyen pour 2 experts différents. La raison est que le calcul du rappel et de la précision comptabilise comme non pertinent à 100 % un choix différent entre le système et l'expert, alors qu'une telle différence de sélection peut n'avoir que de faibles répercussions sur le résumé produit. Cela est dû au fait que les constituants n'ont pas une importance booléenne, mais graduelle. Par exemple, si, dans la phrase de l'exemple 5.1, le système sélectionne juste *la 36e édition du Forum économique mondial de Davos* alors que l'expert sélectionne juste *mercredi 24 janvier à Davos*, alors précision et rappel sont nuls, alors que les deux phrases obtenues après ces deux compressions différentes constituent toutes deux des compressions acceptables.

Exemple 5.1 *Le gratin mondial de l'économie et de la politique ouvre, mercredi 24 janvier à Davos, une station des Alpes suisses, la 36e édition du Forum économique mondial de Davos.*

C'est pourquoi nous nous sommes ensuite orienté vers une évaluation manuelle par notation humaine des compressions produites.

Nous décrivons maintenant les propriétés souhaitées pour notre corpus de documents pour l'évaluation, section 5.3.2.1, puis notre méthode de notation des compressions, section 5.3.2.2.

5.3.2.1 Constitution d'un corpus de documents adéquat pour l'évaluation

Nous présentons maintenant les caractéristiques que doit vérifier notre corpus selon notre protocole d'évaluation. Le corpus choisi pour notre évaluation, selon les critères détaillés dans cette section, sera présenté dans la partie dédiée à l'expérimentation, section 5.4.1.

Le choix du corpus a été principalement orienté par 3 facteurs :

1. la cohérence discursive ;
2. le genre des textes ;
3. la taille des textes.

Un quatrième facteur implicite mais important sur le plan de la démarche scientifique est la mise à l'écart, pour cette évaluation, des corpus utilisés lors du développement de notre approche théorique et conceptuelle. Ainsi, les textes que nous avons choisis pour l'évaluation n'ont jamais été utilisés dans notre travail avant l'évaluation.

La cohérence discursive. Dans l'évaluation de K&M, ce sont des phrases piochées au hasard dans le corpus qui sont compressées par le système puis notées par les juges. Le corpus à noter n'est donc pas un texte cohérent, mais un ensemble de phrases indépendantes. Les juges ne peuvent donc pas s'appuyer sur un discours construit et cohérent pour déterminer quels sont les résumés qui constituent de bons représentants du texte source. Les évaluateurs doivent au contraire considérer chaque phrase indépendamment des autres, pour estimer la qualité des compressions, ce qui est une tâche bien moins naturelle pour l'humain. Afin de faciliter le travail de l'évaluateur, nous avons préféré lui proposer des textes compressés, plutôt que des phrases, pour la notation. Nous discutons de la granularité des éléments à noter dans ces textes en section 5.3.2.2.

Le genre des textes. Comme nous l'avons vu en section 3.5, le genre semble avoir une influence certaine sur l'importance donnée par la fonction syntaxique aux constituants. Afin d'évaluer cette influence, nous avons composé notre corpus de 3 genres différents :

- narratif, pour lequel la compression de phrases semble adéquate ;
- scientifique, qui semble être un plus mauvais candidat à la compression de phrases, ce que nous souhaitions confirmer ou infirmer par l'évaluation ;

– journalistique, qui est un genre souvent considéré en TALN pour les évaluations et auquel l'humain est très régulièrement confronté.

Ces 3 genres ne représentent qu'une petite partie de l'ensemble de ceux existants, le but n'est donc pas d'être exhaustif sur ce paramètre, mais plutôt d'éprouver notre méthode de compression au changement de genre.

La taille des textes. La préparation de l'évaluation ainsi que notre protocole manuel impose une taille des textes relativement petite. En effet, sur de longs textes, le travail de compression puis de notation des compressions serait trop fastidieux pour que nous trouvions un nombre suffisant d'évaluateurs pour notre expérimentation. La qualité d'analyse est aussi un facteur limitant pour la taille du corpus. Ce critère n'entre pas dans la définition du protocole d'évaluation proprement dit, mais il est suffisamment contraignant pour que nous en tenions compte. SYGFRAN fournit une analyse partielle sur environ 70 % des phrases, et sur les 30 % d'analyses correctes, l'attachement des compléments n'est pas toujours juste vis-à-vis de notre grammaire, il est donc important de corriger l'analyse syntaxique afin de ne pas obtenir une forte dégradation de nos résultats. La correction de l'analyse est une tâche assez fastidieuse, elle ne peut alors pas s'effectuer sur un gros volume de textes dans un temps acceptable, dans le cadre de notre expérimentation.

De courts textes ne peuvent être de bons représentants de leur genre, au sens général, cependant, encore une fois, notre but n'est pas l'exhaustivité. Les résultats devront donc être considérés comme indicatifs plutôt que représentatifs.

5.3.2.2 La notation des compressions

Nous présentons maintenant notre protocole de notation des compressions. Nous décrivons d'abord les différents types de compression qui seront notés, puis comment ces compressions sont présentées aux évaluateurs, et enfin comment ces derniers doivent les noter.

Les types de compression. Notre compresseur disposant de 2 modes de production, semi-automatique et automatique, il nous faut noter les compressions issues des deux. Comme le font K&M, nous notons aussi les compressions manuelles afin de comparer leurs notes avec celles obtenues par les compressions issues de COLIN. Ainsi les compressions manuelles ont une double utilité dans notre évaluation : comparer leur temps de production avec celles du mode semi-automatique et leur qualité avec celles des modes automatique et semi-automatique.

Dans la version automatique, nous conservons systématiquement les constituants appartenant à la catégorie d'importance probable élevée, et nous sélectionnons systémati-

quement ceux de la catégorie d'importance probable faible. Concernant la catégorie intermédiaire, d'importance variable, nous avons souhaité tester l'importance de certaines de ses classes de constituants : les modifieurs de la proposition. Ces constituants couvrent en moyenne un pourcentage considérable des mots de la phrase, ce qui leur confère un impact important sur le taux de compression. Sygfran est en mesure de déterminer avec une assez bonne fiabilité les modifieurs de la proposition de type circonstants de temps et de lieu. Ce sont donc ces classes que nous avons considérées.

Nous avons défini 4 types de production de compressions automatiques pour notre évaluation, déterminés par le choix des classes de constituants à supprimer dans la catégorie d'importance variable :

1. suppression de tous les modifieurs circonstanciels de la proposition (MCP) ;
2. suppression des MCP ni de lieu, ni de temps ;
3. suppression uniquement des MCP de lieu ;
4. suppression uniquement des MCP de temps.

Ainsi, pour chaque document original, 6 types de compression seront générés : un manuel, un semi-automatique et 4 automatiques.

La présentation des compressions à noter. Comme nous en avons discuté dans la première section de ce chapitre, notre but est de faciliter la tâche des évaluateurs, pour lesquels nous souhaitons alléger l'effort cognitif et le temps passé à évaluer. Alors que K&M disposent, pour la notation, de phrases indépendantes compressées, nous disposons de textes compressés complets. Les phrases indépendantes sont, grâce à leur petite taille, plus facilement comparables en surface (présence ou absence de mots, cohérence grammaticale), mais plus difficilement sur leur contenu important, lequel dépend en partie d'éléments informationnels présents dans le reste du document. Afin de conserver l'avantage de la petite taille tout en disposant d'un texte complet et cohérent, nous pourrions présenter les textes complets aux évaluateur, mais en les découpant phrase par phrase, et pour chaque phrase proposer à la notation un ensemble de ses compressions. Cependant, l'évaluateur aurait à noter un très grand ensemble de phrases.

Nous avons adopté un choix intermédiaire, où les documents sont découpés en paragraphes. Ainsi, chaque évaluateur doit effectuer un ensemble de notations : pour chaque paragraphe du document original, un ensemble de versions compressées sera présenté à la notation à l'évaluateur. Ce dernier pourra visualiser le paragraphe original ainsi que les différentes compressions avant de procéder à la notation de chaque paragraphe compressé.

Pour un paragraphe donné, 4 compressions automatiques sont disponibles, ainsi qu'un nombre variable de compressions manuelles et semi-automatiques, selon le nombre d'évaluateurs ayant participé à l'étape de compression. Afin de ne pas proposer, pour chaque

paragraphe original, trop de paragraphes compressés à noter, ce qui serait cognitivement éprouvant, et afin d'équilibrer les notations sur les différents modes de compression, nous avons décidé de proposer 3 compressions par paragraphe original, avec une par mode de compression, piochée aléatoirement dans l'ensemble de celles disponibles. Comme dans l'approche de K&M, l'ordre des paragraphes compressés est aléatoire et leur mode de production non précisé.

Le tableau 5.1 illustre notre présentation des compressions par un exemple sur un document composé de trois paragraphes.

P originaux	P compressés		
P_1	CM de P_1	CA type 3 de P_1	CS de P_1
P_2	CS de P_2	CA type 1 de P_2	CM de P_2
P_3	CA type 4 de P_3	CM de P_3	CS de P_3

Tab. 5.1 – Exemple de présentation de compressions à noter pour un document.

Avec P pour paragraphe, P_i pour le ième paragraphe du document, CA pour compression automatique, CS pour compression semi-automatique, CM pour compression manuelle. Dans l'évaluation, chaque cellule de la première colonne contient le texte du paragraphe désigné, et chaque cellule des autres colonnes contient un des paragraphes compressés qui vérifie la condition de la cellule. Par exemple, « CM de P_1 » est une condition qui signifie « être une compression manuelle du premier paragraphe ». L'évaluateur devra noter ici les 9 paragraphes compressés.

Les objectifs sont d'obtenir un maximum de notations pour les 3 modes de production, puis un maximum de notes pour chaque paragraphe compressé. Ainsi, il sera nécessaire de bien répartir l'ensemble des paragraphes compressés à chaque évaluateur, afin de maximiser ces deux objectifs.

La notation du contenu et de la cohérence. Nous avons aussi orienté la notation des paragraphes pour faciliter la tâche de l'évaluateur. Nous ne reprenons pas le découpage de la notation en deux étapes de K&M (une sur le contenu et une sur la grammaticalité des phrases), et préférons nous orienter vers un jugement plus naturel pour l'humain, en demandant à l'évaluateur, pour chaque paragraphe compressé, s'il constitue un bon représentant de l'original. Les notes s'échelonnent de 1 à 5, 1 pour la plus mauvaise note, 5 la meilleure. La consigne suivante a été donnée aux évaluateurs : « Une bonne note doit être donnée à un paragraphe qui conserve les informations les plus importantes du paragraphe original, tout en restant compréhensible et cohérent, une mauvaise dans le cas contraire. »

Nous considérons qu'un résumé qui ne convient pas à un évaluateur n'est pas forcément

un mauvais résumé, mais peut correspondre à deux cas bien différents :

- le résumé ne plaît pas pour des raisons de préférences personnelles, c'est-à-dire que l'évaluateur l'aurait résumé différemment, mais il conçoit que cette compression puisse constituer un résumé possible du paragraphe original ;
- le résumé n'est pas cohérent, c'est-à-dire que l'évaluateur ne parvient pas à interpréter le sens correct de la compression pour des raisons de cohérence textuelle.

Ainsi, lorsque les évaluateurs attribuent une mauvaise note (1 ou 2) à un paragraphe, nous leur demandons de spécifier dans quels cas ils sont : le premier (« Je n'aime pas le résumé ») ou le second (« Je trouve le résumé incohérent »).

Enfin, afin d'aider les évaluateurs à bien interpréter la notation, nous leur fournissons une description pour chaque valeur de note comme ceci :

- note = 1 : la compression ne vous plaît pas ou est incohérente (vous avez à spécifier l'un ou l'autre) ;
- note = 2 : la compression vous plaît peu ou est peu cohérente (vous avez à spécifier l'un ou l'autre) ;
- note = 3 : la compression est assez satisfaisante ;
- note = 4 : la compression est satisfaisante ;
- note = 5 : la compression est très satisfaisante.

5.3.3 Le système d'évaluation

Nous présentons maintenant notre système informatique d'évaluation. Nous commençons par une description rapide des technologies utilisées, puis nous présentons une vue globale des étapes du système.

5.3.3.1 Le système informatique

Afin de faciliter la participation des évaluateurs, nous avons développé le système d'évaluation autour d'une interface Web accessible sur internet.

Les informations utilisées dans l'évaluation sont sauvegardées dans une base de données MYSQL. Voici les tables MYSQL utilisées ainsi que leurs principaux champs (les identifiants ne sont pas précisés) :

- Utilisateur : nom, prénom, adresse mail, tâche affectée (compression manuelle, semi-automatique ou notation), ...
- Document : titre, texte, genre, ...
- Compression : références document et utilisateur, mode de compression, texte compressé, temps de compression, satisfaction d'interaction (pour le mode semi-automatique), commentaire sur la compression, ...

- Paragraphe : référence document ou compression, texte, indice (nième paragraphe), ...
- Note : références paragraphe et utilisateur, note, type d'insatisfaction (si note < 3), commentaire sur la notation, ...

Le langage de programmation utilisé pour générer les pages Web du système est PHP. Le programme PHP permet également de gérer les étapes de l'évaluation, les formulaires HTML, et l'interaction des différents services nécessaires : serveur SYGMART, serveur MYSQL et serveur Apache.

5.3.3.2 Les étapes de l'évaluation

La figure 5.1 illustre les principales étapes du système : celles vues par l'utilisateur (à gauche), et celles réalisées en interne (à droite).

Première étape : compression des documents. Elle commence par l'inscription de l'évaluateur. Le fait de s'enregistrer lui permet de réaliser l'évaluation en plusieurs fois, selon ses disponibilités, et de garder le même identifiant pour les deux étapes. Dès l'inscription, le système va affecter à l'évaluateur une tâche qui est soit celle de compression manuelle, soit celle de compression semi-automatique. Cette affectation se fera en alternance, pour chaque nouvel inscrit, afin de répartir équitablement l'ensemble des évaluateurs sur les deux modes de compression. Puis le système va affecter à l'évaluateur 5 documents à compresser. Ce nombre a été défini d'après la taille et le nombre de nos textes dans notre corpus d'évaluation, afin de proposer un travail raisonnable pour l'évaluateur et suffisant pour l'évaluation. Au moment de l'affectation, sont piochés les documents qui ont été les moins affectés pour le mode de compression concerné.

La compression peut ensuite commencer. Les évaluateurs qui la réalisent de manière manuelle disposent d'un champ texte HTML (élément `textaera`) dans lequel le texte à compresser est affiché, le tout contenu dans un formulaire HTML. Ils peuvent alors parcourir le texte et supprimer les mots qu'ils désirent, avant de valider le formulaire.

Les autres évaluateurs sont dirigés vers l'interface Web de COLIN où ils peuvent compresser leurs textes affectés par interaction de notre outil, puis valider, une fois qu'ils sont satisfaits du résultat. Ces derniers commencent par un tutoriel, basé sur un texte hors corpus, avec lequel ils peuvent s'exercer pour intégrer le fonctionnement de l'interaction. Suite à chaque compression semi-automatique, une demande de satisfaction sur l'interaction est proposée à l'évaluateur, sous la forme d'une notation entre 1 et 5.

Les deux interfaces affichent le temps écoulé et disposent d'un bouton HTML de pause. Les évaluateurs qui ont compressé leurs 5 documents, peuvent demander au système de leur en réaffecter 5 nouveaux, et ainsi de suite jusqu'à épuisement des documents. Le

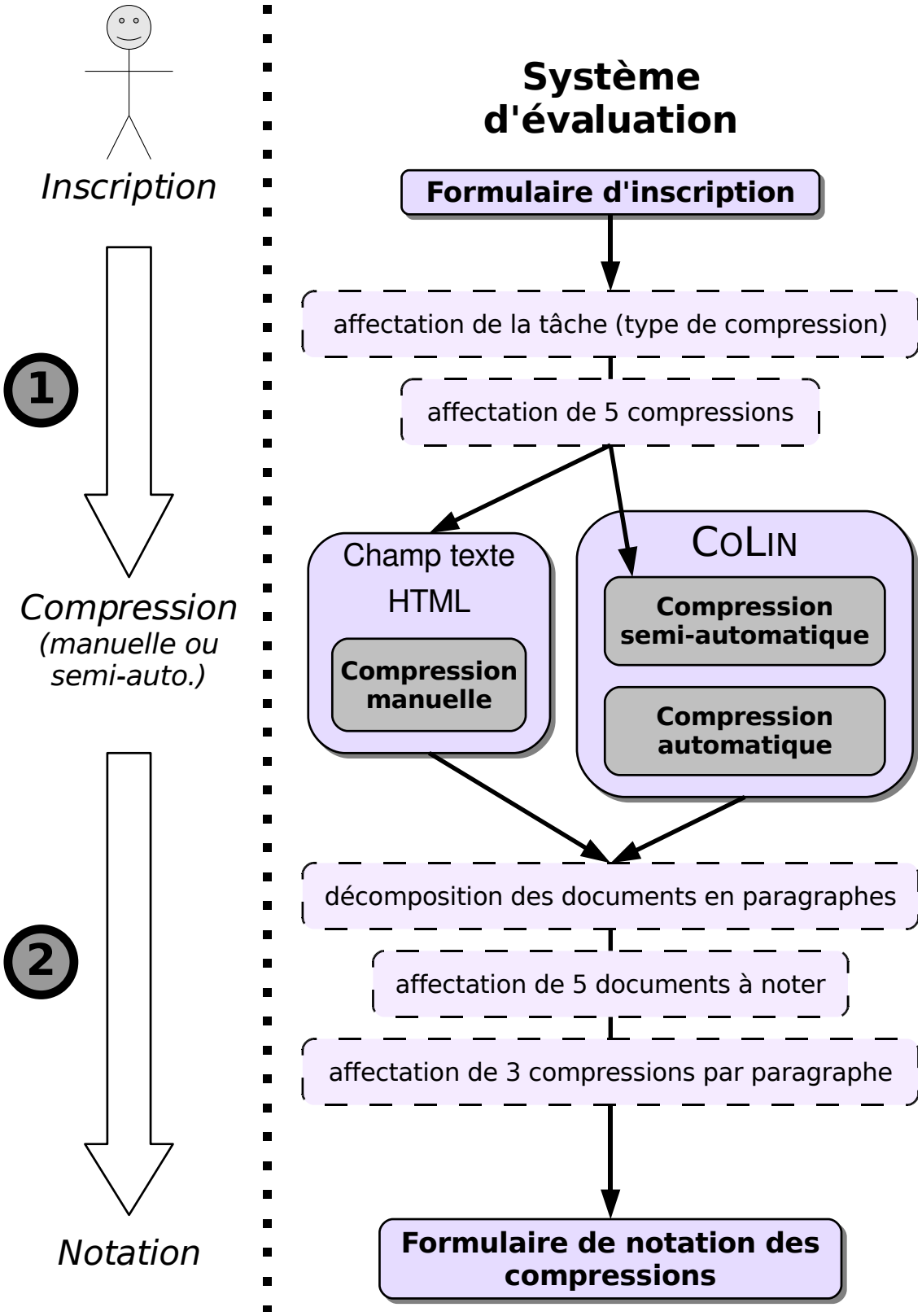

Fig. 5.1 – Étapes de l'évaluation pour l'utilisateur et le système.

système réalise aussi les compressions automatiques en interne.

Nous précisons qu'aucune phrase ne doit être supprimée dans son intégralité, afin de respecter notre protocole. Ainsi il est explicitement spécifié aux évaluateurs qui compressent manuellement de conserver toutes les phrases. Les autres évaluateurs ne peuvent jamais supprimer une phrase complète avec l'interface de CoLin.

Seconde étape : notation des paragraphes compressés. Les compressions se poursuivent jusqu'à une date butoir de commencement de la seconde étape. Une fois cette date atteinte, la première étape n'est plus accessible aux évaluateurs qui sont automatiquement dirigés vers la seconde lors de leur connexion. Le système prépare la seconde étape en réunissant l'ensemble des compressions (des 3 modes) et en les décomposant en paragraphes. Les documents originaux sont aussi décomposés en paragraphes, afin d'être aussi affichés pour la notation. Les nouveaux inscrits commencent directement à la seconde étape.

Puis, chaque évaluateur se voit affecter 5 documents à noter. Pour chaque paragraphe original de chaque document, 3 paragraphes compressés, un de chaque mode de compression, sont piochés parmi l'ensemble des disponibles, issus de la précédente étape. Ces triplets de paragraphes sont affichés alignés avec leur paragraphe original, dans l'ordre des indices de paragraphes. Ainsi l'évaluateur retrouve le document original en parcourant dans l'ordre les paragraphes originaux. Cela peut l'aider à mieux interpréter le sens de l'énoncé et ainsi à mieux juger la qualité des compressions. Ici aussi, une fois les 5 documents notés, l'évaluateur peut en demander davantage à noter.

À la fin de l'évaluation, toutes les données de l'expérimentation sont contenues dans la base de données MYSQL. Nous avons alors procédé à des requêtes SQL pour extraire les informations utiles, puis nous avons utilisé un tableur pour réaliser les calculs nécessaires et produire les différents diagrammes présentés dans la prochaine section.

Captures d'écran. Nous proposons maintenant quelques captures d'écran de l'interface Web du système d'évaluation.

Connexion sous : [redacted]@gmail.com

Système d'évaluation - Résumé semi-automatique - Mehdi Yousfi-Monod

Déconnexion

Étape 1 : Compression de texte

S'entraîner sur le texte de test

Titre	État
Le Monde ; Darfour : la Cour pénale internationale désigne les criminels de guerre	*Fait*
Le Monde ; M. Sarkozy joue l'électorat FN pour le second tour	À faire
Le Monde ; Zimbabwe : le chef de l'opposition présenté devant la justice	À faire
Le Monde ; Forts soupçons de toxicité sur un maïs OGM	À faire
Le Monde ; Les géants de l'informatique s'allient pour lutter conte les e-déchets	À faire

Démarrer Déconnexion

Documentation Contact : [redacted]

Fig. 5.2 – Page d'information sur la progression dans la tâche de compression.

Connexion sous : [redacted]@gmail.com

Système d'évaluation - Résumé semi-automatique - Mehdi Yousfi-Monod

Déconnexion

Compression manuelle du texte "Extrait scientifique 1"

Temps : 23 Pause

Veuillez compresser les phrases de ces paragraphes en supprimant des mots.

Si l'on possède une description des données par un ensemble de D attributs, le problème de la sélection d'attributs consiste à chercher un sous-ensemble de d attributs qui préserve au mieux les informations nécessaires à l'algorithme d'apprentissage. Cette technique sera de nouveau évoquée un peu plus loin au paragraphe [REFERENCE] à l'occasion de la distinction entre filter_methods et wrapper_methods.

Au fond, on est à peu près dans la même situation que celle du réglage des paramètres d'un algorithme (voir le paragraphe [REFERENCE]) : si l'on considère l'algorithme d'apprentissage comme paramétré par le sous-espace de représentation choisi, la question est de trouver le meilleur compromis entre la complexité, mesurée ici par la valeur de d, et l'efficacité, qui est la performance de l'algorithme dans l'espace de dimension réduite de D à d.

La seconde difficulté est qu'il y a un grand nombre de sous-ensembles d'attributs de dimension donnée d et au total 2^D. Il est hors de question de mesurer sur chacun un critère de séparabilité ou la mesure de performance d'un algorithme particulier. On pourrait penser que la structure particulière de cet espace (l'ensemble des sous-ensembles d'un ensemble fini) permet d'utiliser des méthodes approximatives efficaces, mais il faut être prudent à ce sujet, comme le montre l'exemple qui suit.

Considérons le problème d'apprentissage de règle de classification sur un ensemble de cinq points en dimension D=3 donné à la figure [REFERENCE]. Il est facile de voir que les deux classes (représentées par les symboles et o) sont bien séparées, au moins sur cet ensemble d'apprentissage.

Valider la compression Annuler la compression Rétablir le texte initial

Fig. 5.3 – Interface de compression manuelle (allégée d'une partie du texte).

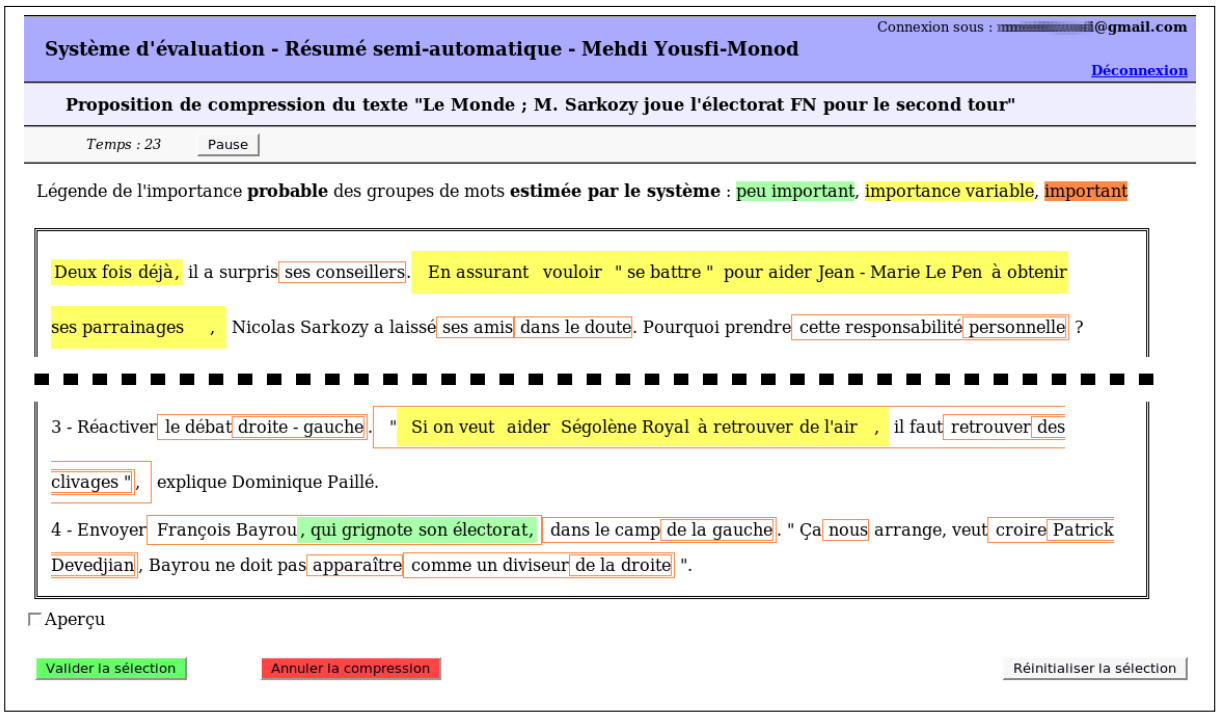

FIG. 5.4 – Interface de compression semi-automatique (allégée d'une partie du texte).

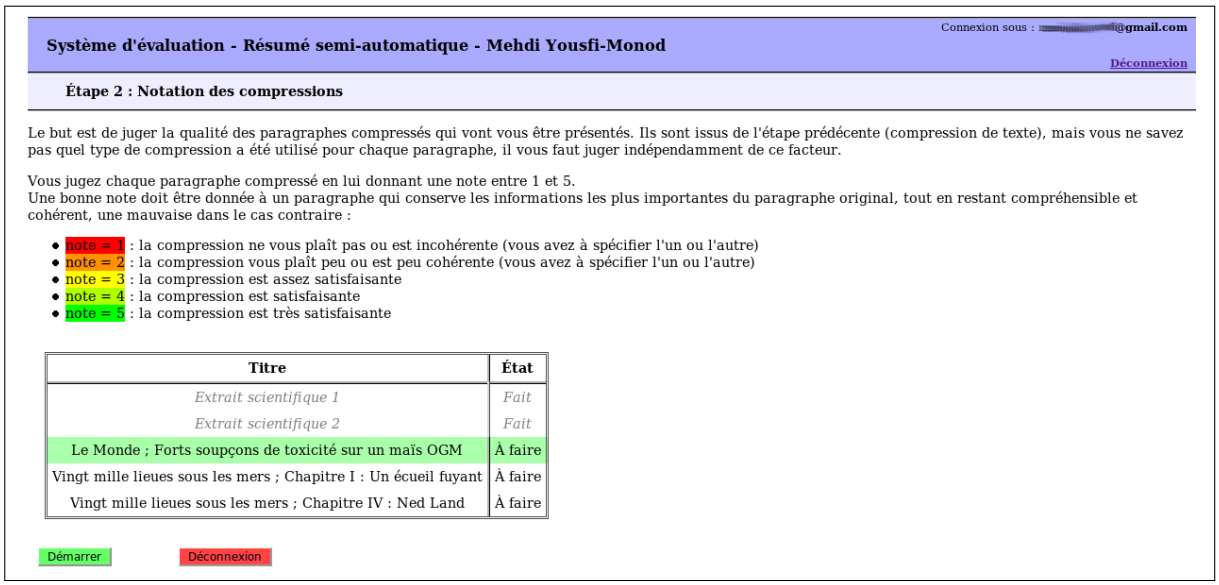

FIG. 5.5 – Page d'information sur la progression dans la tâche de notation.

Paragraphe original	Compressions à noter		
Autorisé à la mise sur le marché en France et en Europe, le MON 863, un maïs transgénique conçu par Monsanto, est depuis plus de deux ans au centre d'une polémique sur son innocuité (Le Monde du 23 avril 2004). Ces débats pourraient reprendre après la publication, mardi 13 mars, dans la revue Archives of Environmental Contamination and Toxicology, d'une étude suggérant une toxicité de cet organisme génétiquement modifié (OGM) pour le foie et les reins.	Le MON 863 est depuis plus de deux ans au centre d'une polémique sur son innocuité. Ces débats pourraient reprendre après la publication, mardi.	Autorisé à la mise sur le marché en France et en Europe, le MON 863, est depuis plus de deux ans au centre d'une polémique sur son innocuité. Ces débats pourraient reprendre après la publication, dans la revue Archives of Environmental Contamination and Toxicology, d'une étude suggérant une toxicité de cet organisme génétiquement modifié pour le foie et les reins.	Autorisé à la mise sur le marché, le MON 863, un maïs transgénique conçu par Monsanto, est au centre d'une polémique sur son innocuité. Ces débats pourraient reprendre après la publication, d'une étude suggérant une toxicité de cet organisme génétiquement modifié.
	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5
Pourquoi une mauvaise note ?	Je n'aime pas C'est incohérent	Je n'aime pas C'est incohérent	Je n'aime pas C'est incohérent
Selon ces travaux, la consommation de maïs MON 863 perturbe plus ou moins fortement, chez le rat, de nombreux paramètres biologiques : poids des reins, poids du foie, taux de réticulocytes (jeunes globules rouges), de triglycérides, etc. La chimie urinaire est également modifiée, avec des réductions de sodium et de phosphore excrété pouvant aller jusqu'à 35 %. Les effets varient selon le sexe des animaux.	Selon ces travaux, la consommation de maïs MON 863 perturbe chez le rat, de nombreux paramètres biologiques. La chimie urinaire est modifiée, avec des réductions de sodium et de phosphore excrété. Les effets varient selon le sexe des animaux.	La consommation de maïs MON 863 perturbe de paramètres biologiques : poids des reins, poids du foie, taux de réticulocytes, de triglycérides. La chimie urinaire est modifiée Les effets varient selon le sexe.	Selon ces travaux, la consommation de maïs MON perturbe de nombreux paramètres biologiques : poids des reins, poids du foie, taux de réticulocytes, de triglycérides. La chimie urinaire est également modifiée, avec des réductions de sodium et de phosphore excrété pouvant aller jusqu'à 35 %. Les effets varient selon le sexe des animaux.
	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5
Pourquoi une mauvaise note ?	Je n'aime pas C'est incohérent	Je n'aime pas C'est incohérent	Je n'aime pas C'est incohérent
"Chez la femelle, on observe une augmentation des graisses et du sucre dans le sang, une augmentation du poids du corps et du poids du foie par rapport au poids du corps, le tout associé à une plus grande sensibilité hépatique, dit M Séralini, principal auteur de cette étude et par ailleurs président du Comité de recherche et d'information indépendantes sur le génie génétique (Criigen). Chez le mâle, c'est le contraire, avec une chute du poids du corps et des reins."	"Chez la femelle, on observe une augmentation des graisses et du sucre dans le sang, une augmentation du poids du corps et du poids du foie par rapport au poids du corps, le tout associé à une plus grande sensibilité hépatique, dit M Séralini. Chez le mâle, c'est le contraire.	" Chez la femelle, on observe une augmentation des graisses et du sucre, une augmentation du poids du corps et du poids du foie, le tout associé à une sensibilité hépatique, dit M Séralini. Chez le mâle, c'est le contraire. "	" Chez la femelle, on observe une augmentation des graisses et du sucre, une augmentation du poids du corps et du poids du foie par rapport au poids du corps, le tout associé à une plus grande sensibilité hépatique, dit M Séralini. C'est le contraire, avec une chute du poids du corps et des reins. "
	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5

FIG. 5.6 – Extrait de l'interface de notation.

5.4 L'expérimentation

Notre protocole étant défini, nous décrivons maintenant l'expérimentation. Nous commençons par présenter le corpus d'évaluation, en fournissant quelques statistiques sur le volume et la répartition du texte, section 5.4.1, puis nous abordons le déroulement de l'expérimentation, section 5.4.2, en détaillant les conditions initiales, les résultats et notre étude de ces derniers.

5.4.1 Le corpus de l'évaluation

5.4.1.1 Présentation

Nous avons constitué notre corpus en prenant en compte les différentes contraintes de cohérence, genre et taille spécifiées en section 5.3.2.1. Nous avons sélectionné 5 textes d'environ 400 mots, pour chaque genre choisi : journalistique, narratif et scientifique (des extraits sont disponibles en annexes). Le tableau 5.2 détaille la répartition des paragraphes, phrases et mots de ces 15 documents.

Nous pouvons observer une répartition homogène des paragraphes dans les documents, ainsi que des phrases dans les paragraphes et des mots dans les phrases.

Le corpus journalistique est composé d'articles du quotidien d'information francophone « Le Monde »[80]. Chaque article a été coupé à environ 400 mots, au paragraphe le plus proche. Le corpus narratif est composé de chapitres du roman « Vingt mille lieues sous

[80] Articles extraits du site internet : `http://www.lemonde.fr/`

	Journalistique	Narratif	Scientifique	Moyenne
Comptage des paragraphes				
Moyenne par document	4,2	6	4,8	5
Maximum par document	5	7	6	-
Minimum par document	3	4	4	-
Comptage des phrases				
Moyenne par document	14,8	17,2	15,4	15,8
Maximum par document	25	22	16	-
Minimum par document	11	10	13	-
Moyenne par paragraphe	3,87	3,04	3,28	3,4
Comptage des mots				
Moyenne par document	389,8	376,6	372	379,47
Moyenne par phrase	28,21	23,36	24,43	25,33

TAB. 5.2 – Répartition du texte dans le corpus.

les mers »[81]. Nous avons sélectionné les 5 premiers chapitres, et conservés environ les 400 premiers mots de chacun, comme pour le corpus journalistique. Enfin le corpus scientifique est composé d'extraits du corpus du même genre de la conférence DEFT'06[82]. Les sciences abordées sont les mathématiques et l'informatique.

5.4.1.2 Prétraitement à l'évaluation

Comme discuté dans le précédent chapitre, section 4.3.2.2, la couverture syntaxique de SYGFRAN n'est pas complète, et même pour les phrases correctement analysées d'après sa grammaire, les attachements de certains compléments ne sont pas toujours corrects vis-à-vis de notre grammaire. Afin de disposer, en entrée à notre compresseur, d'une donnée syntaxique au plus juste, nous avons réalisé plusieurs prétraitements à l'évaluation, sur le corpus, COLIN et SYGFRAN.

Intégration des cas syntaxiques des phrases du corpus. La première initiative a été de fournir le corpus au concepteur de SYGFRAN, Jacques Chauché, afin qu'il intègre, à sa grammaire TELESI, les différents cas de syntaxe présents dans les phrases du corpus. Cette tâche, assez fastidieuse, compte tenu du nombre de règles à manipuler, n'a pas pu aboutir complètement dans le temps que nous nous étions fixé pour cette tâche. Toutefois, les corpus journalistiques et narratifs ont pu être analysés correctement à l'issue du travail effectué sur SYGFRAN.

[81] Disponible en version intégrale sur le site Wikisource, `http://fr.wikisource.org/wiki/Jules_Verne`

[82] www.lri.fr/ia/fdt/DEFT06/corpus/donnees.html

Étiquetage morphologique du corpus. Lorsque SYGFRAN ne reconnaît pas un mot du texte, il est possible d'ajouter une balise devant le mot pour en spécifier sa catégorie grammaticale. Nous avons donc étiqueté le corpus de telles balises, pour les mots inconnus, ce qui a permis de corriger certaines analyses restées partielles.

Balises de sous-analyse syntaxique. Lors de nos tests de correction d'analyse, nous nous sommes rendu compte que certaines parties mal analysées de certaines phrases pouvaient être correctement analysées lorsqu'elles étaient extraites puis fournies seules à l'analyseur syntaxique. Cela nous a incité à demander à J. Chauché d'ajouter à SYGFRAN un mécanisme d'analyse indépendante de sous-parties de la phrase. Avec cette fonctionnalité, il nous est maintenant possible d'encadrer un constituant de la phrase par deux balises spécifiques, ce qui a pour conséquence de forcer SYGFRAN à d'abord analyser cette partie, puis d'intégrer le résultat à l'analyse complète. Ce système fonctionne aussi pour les inclusions de balises, permettant alors une grande manipulation de la phrase, pour en améliorer la qualité de son analyse. Ainsi, ces balises nous ont permis de corriger environ 50 % des phrases analysées partiellement.

Règles transformationnelles *ad hoc*. Pour tous les autres cas d'analyses partielles et de mauvais attachement, nous avons décidé créer des règles *ad hoc*, pour chaque phrase problématique, afin d'obtenir une qualité d'analyse du corpus satisfaisante dans un bon délai de temps.

Créer une règle *ad hoc* pour une phrase donnée du corpus se fait facilement en identifiant une partie de sa structure, dont celle que nous souhaitons modifier, ainsi que quelques mots spécifiques à la phrase, de telle sorte que seule cette phrase vérifie ces contraintes.

La correction des analyses partielles a consisté à transformer les phrases concernées par une suppression du nœud racine de la phrase, étiqueté `ULFRA`, puis un attachement correct des constituants, et enfin un renseignement des variables de fonction syntaxique.

La correction des analyses complètes mais avec un attachement de certains compléments incorrect a été réalisée de manière plus générique lorsque cela été possible. Nous avons pour cela créé un ensemble de règles reconnaissant les têtes lexicales impliquées, plutôt que les phrases complètes. L'exemple de code source TELESI présenté dans le chapitre précédent, section 4.3.1.3, illustre de telles règles, toutefois simplifiées pour la présentation de SYGFRAN. Nos règles complètes, accessibles en annexe, tiennent compte de la présence de prépositions et ponctuations, et gèrent précisément le déplacement des branches, vis-à-vis des nœuds déjà existants. Ainsi nous avons créé 1 règle pour un nom, 1 règle pour un adjectif, 1 règle pour une préposition et 41 règles pour des verbes. Par exemple, certaines règles des verbes recherchent les constructions du type *accéder à A*, *agir comme A*, *distinguer A de B*, *être aimable de/pour A*, *se changer en A*. Les deux

informations les plus utilisées pour identifier les compléments des verbes sont le type de circonstant et le lemme de la préposition.

Ces règles de correction de l'attachement des compléments ne sont cependant pas fiables en toute circonstance, car de nombreux cas particuliers de la langue peuvent les mettre en échec. Nous nous sommes assuré qu'elles fonctionnaient correctement pour notre corpus d'évaluation, ce pour quoi elles ont été spécifiquement créées.

5.4.2 Résultats, discussion et bilan

Nous décrivons maintenant les résultats de l'expérimentation sur l'évaluation de CoLin. Nous commençons par fournir des informations concernant la participation et l'activité des évaluateurs. Puis nous donnons les résultats de leur travail ainsi que notre interprétation de ces derniers.

5.4.2.1 Participation

Le niveau de participation des utilisateurs à notre expérimentation joue un rôle important dans la pertinence des résultats de l'évaluation. Il est important de disposer de suffisamment d'évaluateurs actifs pour obtenir suffisamment de compressions par document et de notations par paragraphe. Cette section présente les différentes valeurs de participation, qui se révèlent être satisfaisantes pour cette tâche.

Utilisateurs. 39 utilisateurs se sont inscrits au système d'évaluation. Ces utilisateurs sont pour une grande majorité des doctorants et docteurs en informatique, en TALN, ou en linguistique computationnelle, maîtrisant tous le français. Nous avons jugé leur niveau de connaissance comme largement suffisant pour saisir correctement le contenu informationnel des textes de notre corpus, ce qui a été confirmé *a posteriori* par les commentaires de ces évaluateurs suite aux compressions et notations réalisées. 14 de ces utilisateurs n'ont cependant pas franchi le stade de l'inscription ou du tutoriel. Les 25 autres ont effectués au moins une compression ou notation. 12 évaluateurs ont participé aux deux étapes de l'évaluation. Le tableau 5.3 résume la participation au système d'évaluation.

	Nb. d'utilisateurs
Total	39
Total actifs (évaluateurs)	25
Actifs Compression	19
Actifs Notation	18
Intersection	12

Tab. 5.3 – Participation à évaluation.

Compressions. Le système d'évaluation répartissant aux mieux les compressions à effectuer, ces dernières s'équilibrent assez bien sur les genres textuels et les modes de compression, comme on peut l'observer dans le tableau 5.4.

	Journalistique	Narratif	Scientifique	Moyenne	Total
Effectif					
Manuel	23	25	11	19,67	59
Semi-automatique	7	15	26	16	48
Automatique	20	20	20	20	60
Moyenne par document					
Manuel	4,6	5	2,2	3,93	-
Semi-automatique	1,4	3	5,2	3,2	-
Automatique	4	4	4	4	-

Tab. 5.4 – Répartition des nombres de textes compressés selon les genres.

Ainsi, à l'issue de la première étape, chaque document disposait d'entre 3 et 4 compressions pour chaque mode, ce qui nous fourni une information importante pour étudier la subjectivité des résumés produits. Les légers déséquilibres sont dus aux évaluateurs qui n'ont pas ou peu réalisé leur tâche, c'est-à-dire compresser au moins 5 documents (il ne nous était pas possible de déterminer à l'avance lesquels la réaliseraient au complet).

Notations. La répartition des notes est davantage équilibrée et fournie que celle des compressions comme l'illustre le tableau 5.5.

	Journalistique	Narratif	Scientifique	Moy.	Total
Effectif					
Manuel	140	107	124	123,67	371
Semi-automatique	140	124	124	129,33	388
Automatique	140	124	124	129,33	388
Moyenne par paragraphe					
Manuel	6,67	3,57	5,17	5,13	-
Semi-automatique	6,67	4,13	5,17	5,32	-
Automatique	6,67	4,13	5,17	5,32	-

Tab. 5.5 – Répartition des nombres de paragraphes notés selon les genres.

La raison est que chaque document noté se compose d'un nombre de notes égal à 3 fois le nombre de paragraphes, c'est-à-dire en moyenne $3*5 = 15$ notes, contre une compression par document pour la première étape. Nous pouvons constater que chaque paragraphe compressé est noté en moyenne plus de 5 fois, nous disposons ainsi de suffisamment d'avis différents pour en tirer une note moyenne significative.

Compressions et notations individuelles. À l'échelle de l'évaluateur, la répartition des compressions et notations est assez variable. Nous pouvons lire dans le tableau 5.6 que 4 évaluateurs ont réalisé au moins 10 compressions, alors que 5 n'ont pas dépassé les 5 compressions.

Effectif	Nb. évaluateurs
Moins de 5	5
Entre 5 et 7	9
Entre 10 et 15	4

TAB. 5.6 – Répartition des nombres de textes compressés selon les évaluateurs.

La majorité se regroupe toutefois dans la tranche des 5–7 compressions ce qui constitue la tâche demandée pour la première étape.

La répartition des notations est aussi assez variée, comme l'illustre le tableau 5.7.

Effectif	Nb. évaluateurs
Moins de 30	4
Entre 30 et 70	10
Entre 70 et 85	3

TAB. 5.7 – Répartition des nombres de paragraphes notés selon les évaluateurs.

Encore une fois, quelques évaluateurs se sont à peine impliqués dans la tâche alors que d'autres ont fourni un travail au-delà du minimum qui leur été demandé. La majorité se situe dans la tranche du travail demandé.

Chaque évaluateur a réalisé en moyenne 6 compressions et 64 notations, ce qui représente une participation significative pour la validité de notre expérimentation.

5.4.2.2 Résultats

Nous abordons enfin les résultats sur les performances de COLIN. Nous présentons l'ensemble des informations que nous avons jugées utiles à notre étude, et discutons de ces résultats.

Le temps de compression. Il est pris en compte pour l'évaluation de l'aide apportée par la version semi-automatique du compresseur par rapport à la compression manuelle. Un temps plus court pour un résumé à taux et qualité de compression égaux ou supérieurs signifie une aide accrue.

Nous ne considérons pas ici le temps de compression de la version automatique, lequel est négligeable par rapport aux 2 autres. En effet, la compression automatique, sur une

machine relativement récente (microprocesseur cadencé à 2 Ghz), demande environ 5 secondes de traitement par texte du corpus, alors que la compression pour les 2 autres modes requiert des temps se comptant en centaines de secondes.

Le tableau 5.8 présente les temps moyens de compression, par genre et modes de compression.

	Journalistique	Narratif	Scientifique	Moyenne
Moyen en manuel	422	289	267	326
Moyen en semi-auto.	273	313,4	220,5	268,97
Gain en semi-auto.	35,31 %	-8,44 %	17,42 %	17,49 %

TAB. 5.8 – Temps de compression d'un document, en secondes.

Le gain de temps en pourcentage de la compression semi-automatique sur la manuelle est affiché en troisième ligne. En moyenne ce gain s'élève à presque 20 %, la compression semi-automatique prend donc moins de temps que la manuelle.

Selon le genre, le gain varie considérablement. Le plus flagrant concerne le temps de compression des textes du genre journalistique qui a été grandement amélioré grâce à l'interface de COLIN. La mise en valeur des constituants effaçables semble avoir été efficace pour ce genre. Le seul gain négatif concerne le corpus narratif. L'apport de l'interface interactive a-t-il incité les évaluateurs à davantage réfléchir sur l'importance des différents constituants ? Face à ces gains variables, une autre question peut se poser : le temps de compression est-il corrélé aux taux et qualité de compression ?

Ces valeurs moyennes ne sont toutefois pas totalement représentatives du réel gain de temps à plus long terme, car il faut un certain temps avant de bien s'habituer au fonctionnement de l'interface interactive. Le tutoriel a permis aux évaluateurs de se familiariser avec l'outil, mais n'est pas suffisant pour les entraîner jusqu'à un niveau d'utilisation optimal. Le diagramme de la figure 5.7 reprend le temps moyen du précédent tableau, et nous renseigne aussi sur le temps moyen pris par les évaluateurs pour réaliser les 4ème et 5ème compressions, ainsi que sur la 5ème uniquement.

Nous avons considéré les moyennes sur les 4ème et 5ème compressions, car ce sont celles qui ont été effectuées en dernier par la majorité des évaluateurs, donc à un niveau d'entraînement optimal dans le cadre de notre évaluation. Le diagramme expose nettement le gain de temps croissant sur les derniers textes compressés. Sur le 5ème, le temps est quasiment divisé par 2 pour les compressions semi-automatiques.

Une autre donnée significative sur le gain de temps est la progression individuelle de chaque évaluateur durant ses propres compressions. Afin de l'estimer, nous avons considéré, pour chaque évaluateur, le temps pris pour le premier texte compressé, puis le

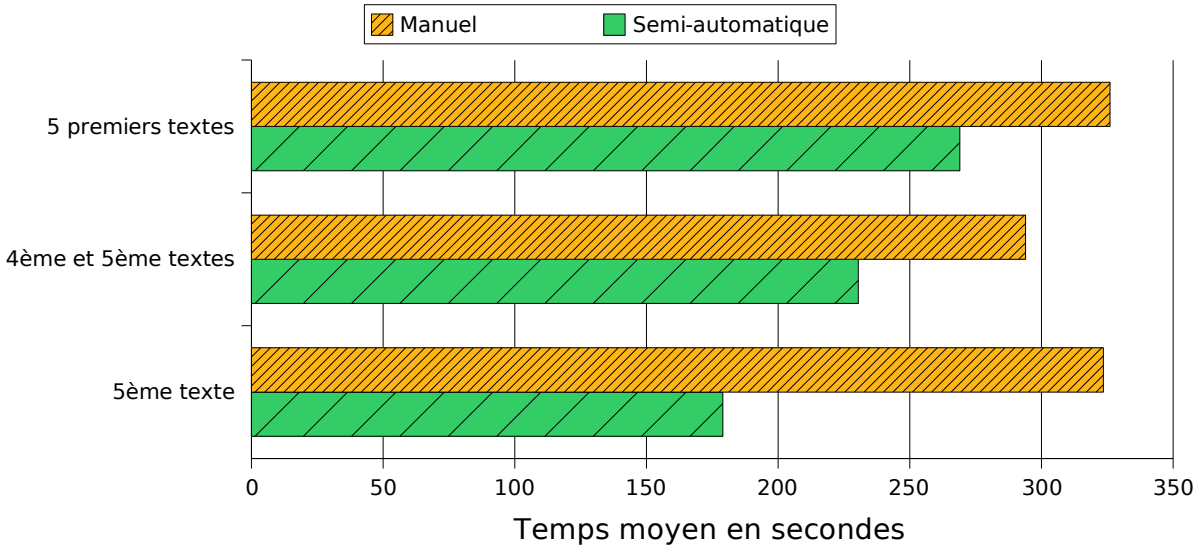

FIG. 5.7 – Temps de compression moyen.

rapport de temps de chaque texte suivant par rapport au premier. Nous avons ensuite effectué une moyenne sur l'ensemble des évaluateurs, pour chaque mode de compression considéré. Le diagramme de la figure 5.8 illustre cette progression.

Le premier texte a un rapport de 1, car le temps est comparé à lui-même. Le second texte voit un rapport accru pour la compression semi-automatique. Nous pensons que c'est dû à la phase de familiarisation à l'outil. Puis le rapport chute considérablement pour ce mode de compression, alors qu'il ne chute que faiblement pour la compression manuelle. L'utilisation de cette dernière s'acquiert donc rapidement et ne propose pas de gain de temps accru à long terme. L'aide apportée par COLIN est donc considérable sur le plan du temps gagné.

Satisfaction de l'interaction avec COLIN. Elle représente le niveau de satisfaction de l'évaluateur suite à l'interaction avec l'interface de notre compresseur et permet de mesurer la qualité de l'interface en termes d'ergonomie et de plaisir d'interaction. Cette satisfaction fut élevée, comme l'illustre le diagramme de la figure 5.9.

Ainsi les évaluateurs ont apprécié se servir de l'interface, ce qui a pour avantage de leur faciliter l'apprentissage et l'utilisation.

Le taux de compression. Il représente le pourcentage du nombre de mots supprimé par rapport au nombre de mots du texte original. Le tableau 5.9 nous renseigne sur le taux des 3 modes de compression, ainsi que le détail pour le mode automatique, selon les types de modifieurs circonstanciels de la proposition (MCP) effacés.

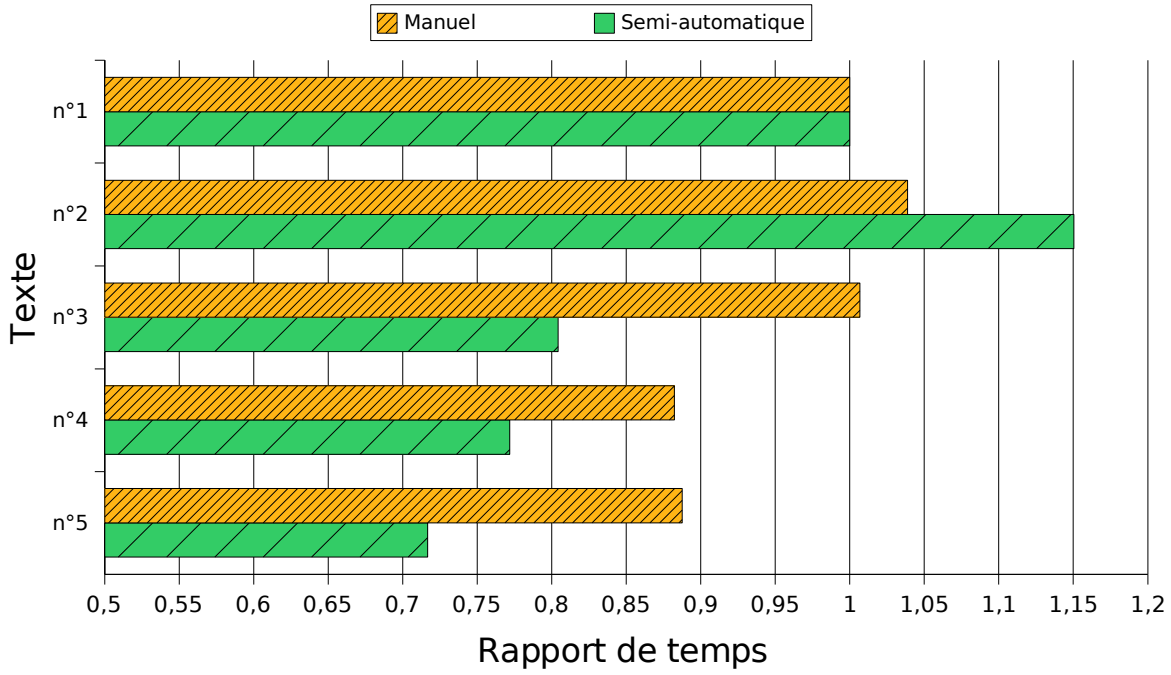

FIG. 5.8 – Progression du temps de compression.

La compression semi-automatique dispose du meilleur taux, dépassant les 40 %, suivi par l'automatique, avec 32 % puis enfin la manuelle avec 25 %. Au niveau du genre textuel, pour les modes manuel et semi-automatique, le corpus journalistique obtient un taux de compression très élevé, et à notre étonnement, les corpus narratif et scientifique ont obtenu des taux assez proches. Nous pensions que le genre scientifique serait beaucoup moins sujet à la compression, cependant, même pour les compressions manuelles, les évaluateurs ont effacé davantage de mots que pour le genre narratif.

Les taux moyens par mode de compression, pour les 3 genres réunis, sont graphiquement représentées dans le diagramme de la figure 5.10.

Ce critère reste tout de même peu significatif en dehors de toute notation de qualité des compressions produites, que nous abordons maintenant.

La notation des compressions. La moyenne des notes affectées aux paragraphes compressés est présentée dans le tableau 5.10.

Les compressions manuelles et semi-automatiques ont obtenu des notes quasiment égales. Sachant que le taux de compression des secondes est nettement meilleur que celui des premières, et que le temps moyen et à plus long terme est aussi à l'avantage du mode semi-automatique, nous concluons que COLIN propose une réelle aide au résumé automatique, à travers son interface Web.

Les compressions automatiques sont naturellement en retrait, cependant, elles restent

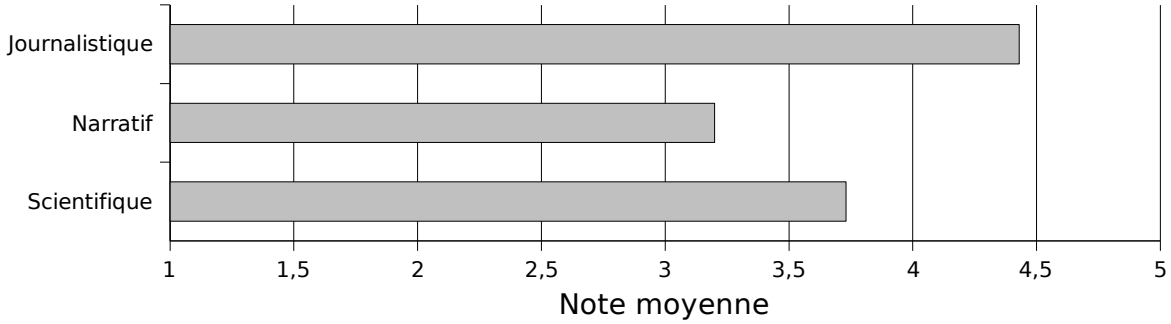

FIG. 5.9 – Notes de satisfaction sur l'interaction avec COLIN.

	Journalistique	Narratif	Scientifique	Moyenne
Manuel	35,62 %	17,08 %	22,97 %	25,23 %
Semi-automatique	48,06 %	37,14 %	38,37 %	41,19 %
Automatique moyen	37,76 %	24,84 %	33,17 %	31,92 %
Détail de l'automatique				
Auto. (1) tous les MCP	45,35 %	42,23 %	46,44 %	44,67 %
Auto. (2) MCP ni lieu, ni temps	37,83 %	34,59 %	40,54 %	37,66 %
Auto. (3) MCP lieu	24,29 %	21,93 %	25,15 %	23,79 %
Auto. (4) MCP temps	20,39 %	17,40 %	20,69 %	19,50 %

TAB. 5.9 – Taux de compression.

proches des 2 autres et avoisinent une qualité assez satisfaisante (note égale à 3). Parmi ces dernières, celles qui ont obtenu les meilleures notes sont les (2). Les modifieurs de la proposition supprimés dans ce type de compression semblent donc constituer les meilleurs candidats parmi les constituants sélectionnables de la catégorie d'importance variable. De plus, le temps de compression du mode automatique est sans comparaison avec les 2 autres et le taux de compression de « Auto. (2) » est meilleur que le manuel et sensiblement inférieur au semi-automatique, nous concluons donc que le mode de compression automatique de COLIN dispose du meilleur rapport de temps/taux/qualité de compression parmi l'ensemble des modes.

Si nous considérons la notation par genre, nous pouvons proposer une réponse à la question du lien entre temps et qualité de compression : les deux semblent bien être corrélés. En effet, pour le genre journalistique, où le temps avait été bien plus court pour le mode semi-automatique, ses compressions sont moins bien notées que celles du mode manuel. De même, le genre narratif vérifie le phénomène, mais dans le sens opposé. Le genre scientifique obtient la même notation pour les 2 premiers modes, ce qui confère au second un léger avantage car son temps de compression avait été inférieur.

Pour le mode « Auto (2) », la notation des genres nous révèle aussi que le corpus

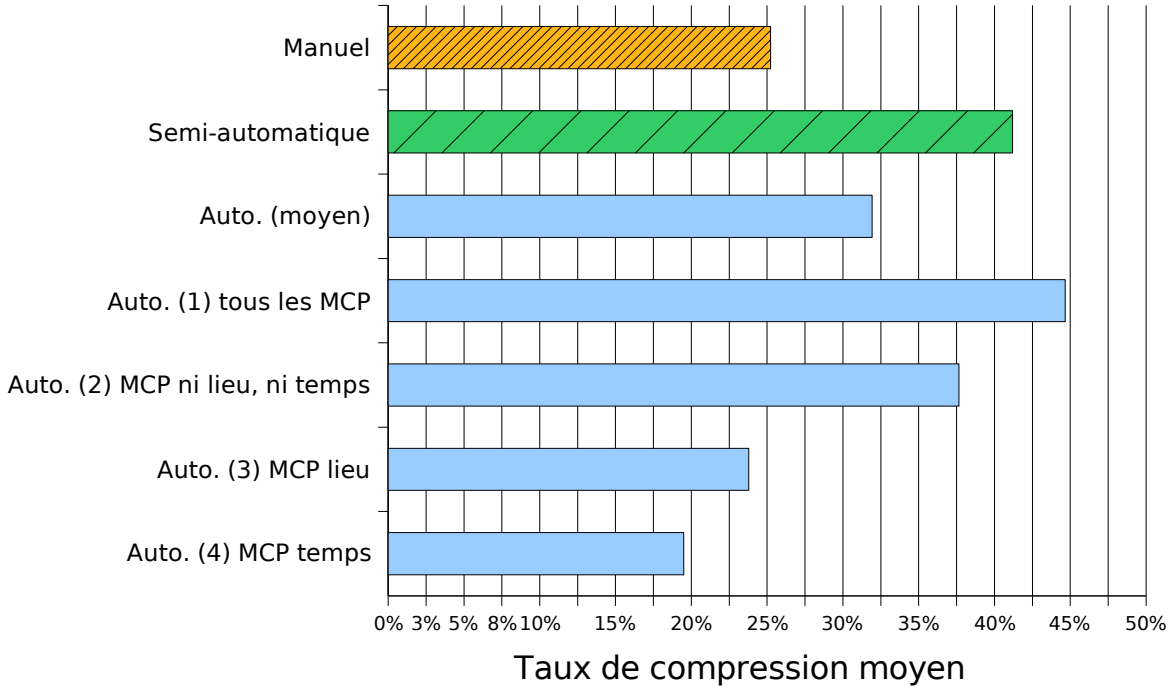

FIG. 5.10 – Taux de compression moyen, basé sur le comptage des mots.

journalistique a bien été compressé, en obtenant une note de 3,7 pour un taux de compression de 38 %, ce qui est un rapport supérieur aux 2 autres genres. Contrairement à nos prévisions sur l'affinité à la compression du genre effectuées au chapitre 3, les genres narratif et scientifique obtiennent étrangement des résultats très proches pour ce mode de compression : une notation égale pour un taux de compression légèrement à l'avantage du scientifique. Par contre, le genre narratif est celui que le mode semi-automatique a le plus avantagé, par rapport au mode manuel, en gains du taux de compression (+20 %) et de la notation (+0,3). Pour ce genre, notre interface a donc permis de bien révéler à l'utilisateur certains constituants peu importants qu'il n'aurait pas forcément considéré.

Nous retrouvons ces notations dans le diagramme de la figure 5.11, avec une moyenne sur les 3 genres.

Nature des mauvaises notes. Lorsqu'une note inférieure à 3 est donnée par un évaluateur, il spécifie si c'est parce qu'il n'est pas d'accord sur les éléments importants effacés (raison de préférence) ou si c'est parce qu'il trouve la compression incohérente (raison de cohérence).

Les tableaux 5.11 et 5.12 renseignent, respectivement, sur les insatisfactions de type préférence et cohérence, selon le genre et le mode compression.

Pour comparer ces valeurs entre elles, nous les rapportons au nombre de paragraphes compressés, pour chaque mode de compression. Le diagramme de la figure 5.12 présente

	Journalistique	Narratif	Scientifique	Moyenne
Manuel	4,03	3,67	3,41	3,7
Semi-automatique	3,61	3,94	3,41	3,65
Auto. (moyen)	3,34	2,72	2,9	2,99
Détail de l'automatique				
Auto. (1) tous les MCP	3,23	2,61	2,77	2,87
Auto. (2) MCP ni lieu, ni temps	3,7	3	3	3,23
Auto. (3) MCP lieu	3,09	2,77	3,11	2,99
Auto. (4) MCP temps	3,52	2,56	2,47	2,85

TAB. 5.10 – Valeur moyenne des notes de paragraphe.

	Journalistique	Narratif	Scientifique	Total
Manuel	5	9	14	28
Semi-automatique	11	7	12	30
Automatique (total)	22	19	10	51
Détail de l'automatique				
Auto. (1) tous les MCP	8	6	4	18
Auto. (2) MCP ni lieu, ni temps	4	2	1	7
Auto. (3) MCP lieu	6	6	1	13
Auto. (4) MCP temps	4	5	4	13

TAB. 5.11 – Nombre de paragraphes jugés insatisfaisants de type *préférence*.

graphiquement les pourcentages d'insatisfaction par paragraphe compressé, pour les 3 modes de compression.

Nous observons qu'il y a environ autant d'insatisfactions préférentielles pour les modes manuels, semi-automatique, et automatique (2)[83], pour une valeur avoisinant les 10 % des paragraphes compressés, ce qui nous incite à penser que l'évaluation de la qualité d'un résumé est subjective, ce qui nous conforte dans notre protocole d'évaluation, où ne sont pas utilisés des résumés de référence pour déterminer la qualité des compressions.

Nous notons aussi que les compressions semi-automatiques sont davantage sujettes à des incohérences que les manuelles. Ces 2 modes laissant à l'utilisateur le choix des suppressions, ce résultat semble étrange. Nous avons alors réalisé une plongée dans les paragraphes compressés, pour étudier quels étaient les cas de tels compressions incohérentes en mode semi-automatique. Trois principales raisons semblent en être la cause :

– dans certains cas, les ponctuations, contractions ou élisions n'ont pas été correctement restituées dans les résumés, ceci dû à un bug dans le script de génération des compressions de COLIN. Ainsi, des phrases se voyaient terminées sans point fi-

[83]Nous considérons ce type de mode car nous avons noté qu'il constituait le meilleur candidat parmi l'ensemble des automatiques.

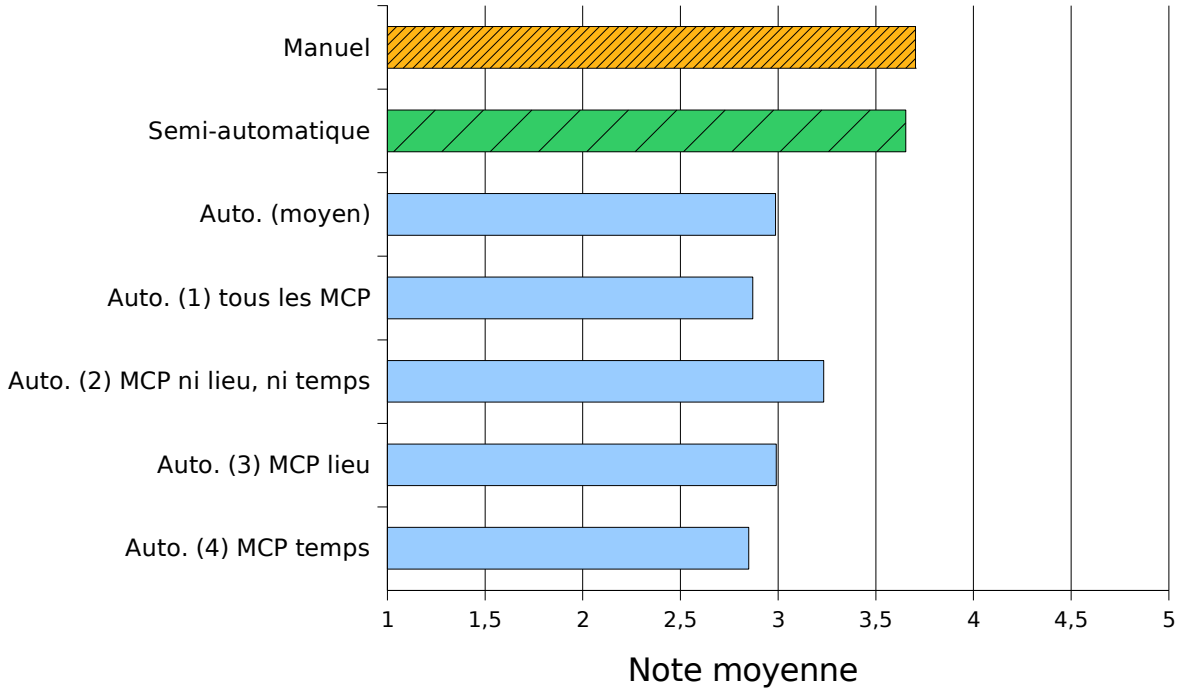

FIG. 5.11 – Valeurs moyennes des notes par paragraphe.

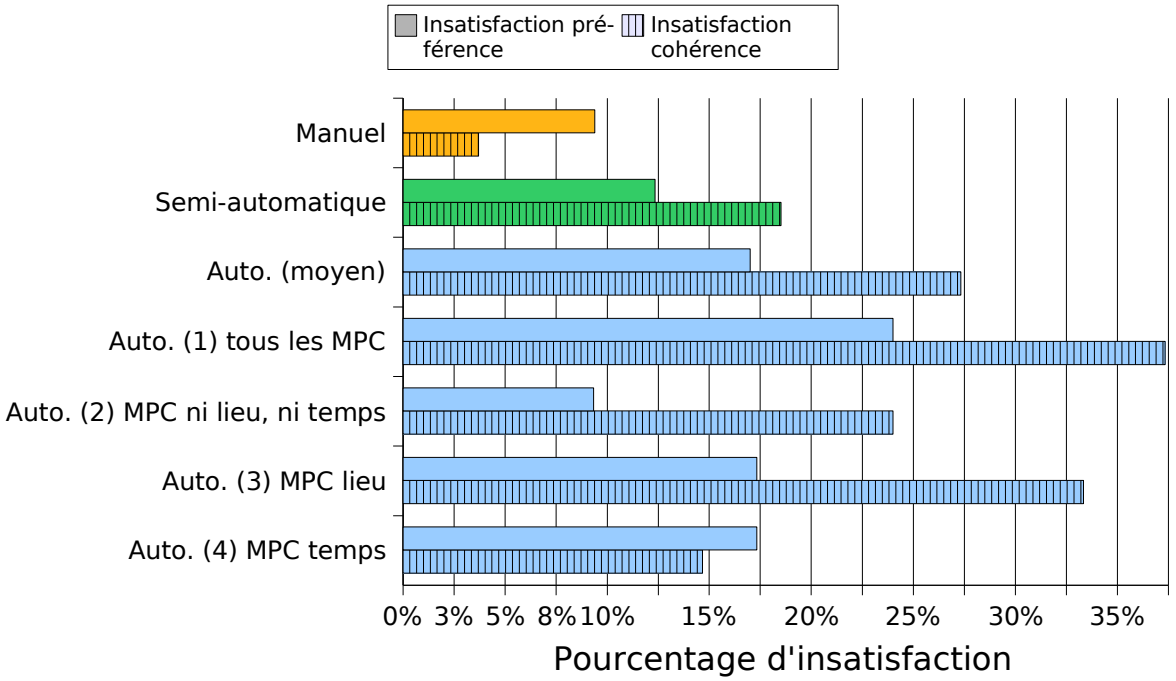

FIG. 5.12 – Nombre et type des insatisfactions sur les mauvaises notes.

	Journalistique	Narratif	Scientifique	Total
Manuel	7	1	3	11
Semi-automatique	17	10	18	45
Automatique (total)	15	28	39	82
Détail de l'automatique				
Auto. (1) tous les MCP	7	9	12	28
Auto. (2) MCP ni lieu, ni temps	1	5	12	18
Auto. (3) MCP lieu	6	9	10	25
Auto. (4) MCP temps	1	5	5	11

TAB. 5.12 – Nombre de paragraphes jugés insatisfaisants de type *cohérence*.

nal, ou alors des virgules importantes étaient supprimées, ou encore certains articles n'étaient pas élidés alors qu'ils auraient dû l'être... ;

– l'analyse syntaxique des phrases n'est pas toujours parfaite, malgré notre prétraitement à l'évaluation. Certaines fonctions syntaxiques restent mal renseignées, et certains attachements de constituants sont incorrects. Cela cause une légère dégradation de nos résultats et pourrait être limité avec une meilleure analyse ;
– certains évaluateurs ont préféré supprimer certains constituants qu'ils jugeaient peu importants sans se soucier de la cohérence du résumé final. Le facteur compression a donc certainement été plus important à leurs yeux que celui de la cohérence.

Pour cette dernière raison, les évaluateurs voulaient très certainement supprimer la proposition ou la phrase complète contenant le constituant effacé. D'ailleurs certains commentaires laissés sur les compressions confirment cette intuition, dans lesquels les évaluateurs expriment leur envie de supprimer certaines phrases complètes. Bien que cela ne fasse pas partie des objectifs de notre approche, la suppression de phrases complètes peut être envisagée dans une extension à notre modèle, dans le but de fournir un résumé à un taux de compression plus élevé.

Les compressions automatiques sont celles qui possèdent le plus d'incohérences, tout en restant dans un même ordre de grandeur que pour les autres modes. Le problème le plus fréquent ici concerne les suppressions de certains modifieurs dont l'apport informationnel est important pour l'interprétation de l'énoncé. Cela se vérifie particulièrement pour les textes scientifiques comme nous l'avions supposé dans nos premières estimations, décrites dans le chapitre 3. De telles dégradations du contenu ont été souvent interprétées comme des cas d'incohérences pour certains évaluateurs.

Enfin, encore une fois, la compression automatique (2) constitue la meilleure candidate parmi les 4 pour les insatisfactions de type préférence, et arrive seconde pour celles de type cohérence.

5.4.2.3 Bilan

Nous rappelons que le but de cette expérimentation a été de valider notre hypothèse de départ qui est que, pour la compréhension d'un texte, la fonction syntaxique des constituants des phrases est un facteur conséquent dans l'évaluation de l'importance de ces constituants. Nous établissons maintenant notre interprétation des résultats de l'expérimentation en regard de cette hypothèse.

La bonne qualité des compressions obtenues par la version automatique de CoLin, en terme de taux de compression comme de conservation de contenu informationnel important, confirme l'influence certaine de la fonction syntaxique dans l'importance des constituants. En effet, supprimer les constituants disposant d'une des fonctions syntaxiques que nous avons identifiées comme peu importantes a permis à CoLin de produire des textes compressés vérifiant, en moyenne, un bon niveau de représentation des textes originaux.

Le choix du constituant comme granularité d'analyse de la phrase s'est révélé particulièrement adéquat, au regard des résultats du mode semi-automatique face à ceux du mode manuel. En effet, d'une part le gain considérable en taux de compression indique que le constituant englobe la majorité des segments textuels qu'un humain souhaite supprimer dans une phrase, et d'autre part le gain considérable de temps de compression indique que le constituant n'est pas une unité textuelle trop petite car elle permet d'être facilement et rapidement manipulable par un être humain, à travers une interface graphique.

5.5 Conclusion

Dans ce chapitre, nous avons décrit l'évaluation de notre approche sur la compression syntaxique de phrases à travers une expérimentation utilisant notre compresseur CoLin. Nous avons commencé par présenter la méthode actuelle la plus répandue pour évaluer les résumés automatiques : ROUGE. Cette méthode, automatique et statistique est peu adaptée au résumé par compression de phrases. De plus, procédant par comparaison des résumés automatiques avec des résumés de référence, la qualité d'évaluation de ROUGE est limitée par la subjectivité intrinsèque des résumés de référence produits par des humains. Notre choix s'est alors orienté vers une évaluation manuelle. Celle de Knight et Marcu, présentant des caractéristiques proches de nos attentes, nous nous sommes appuyé sur cette méthode, en l'adaptant à nos exigences en termes d'allègement de l'effort cognitif imposé aux évaluateurs, afin de motiver ces derniers et de disposer d'un maximum de participation à notre évaluation.

Notre protocole se décompose en deux parties : l'évaluation de l'aide apportée par la version semi-automatique de CoLin et l'évaluation de la qualité des compressions produites par les deux versions du compresseur. Nous mesurons l'aide apportée selon 4

critères : le temps gagné, l'effort cognitif allégé, la satisfaction d'utilisation de l'outil et la qualité des compressions produites accrue. Dans la seconde partie du protocole nous proposons des critères de constitution du corpus d'évaluation, basés sur la cohérence discursive, le genre textuel et la taille des textes, puis nous définissons comment nous souhaitons noter les compressions. Nous décrivons alors les modes de compressions utilisés (manuel, semi-automatique et 4 types d'automatique), puis notre présentation des compressions à l'évaluateur (découpées en paragraphes), et enfin la méthode de notation, qui a pour but d'attribuer une note selon la conservation du contenu important et la cohérence globale des compressions. Nous terminons cette section en présentant le système informatique utilisé pour l'évaluation, des technologies utilisées à la décomposition en étapes et traitement du logiciel.

Enfin nous abordons l'expérimentation. Le corpus utilisé est présenté, ainsi que les prétraitements que nous avons dû effectuer sur le corpus, sur SYGFRAN, et sur COLIN, afin que l'analyse syntaxique du corpus soit d'une qualité suffisamment élevée pour permettre à notre compresseur de disposer des données nécessaires à son bon fonctionnement. Les résultats sont ensuite exposés, en commençant par des informations sur la participation des évaluateurs, leur répartition sur les tâches, les compressions et notations produites, puis en continuant avec les données de temps de compression, de satisfaction d'interaction, de taux de compression, de notation des compressions et de type des incohérences dans le cas des mauvaises notes. La compression interactive de COLIN permet de gagner presque 20 % de temps, par rapport à la manuelle, et ce gain augmente avec l'entraînement des évaluateurs à l'outil interactif, jusqu'à presque 100 % sur le 5ième texte compressé. Les évaluateurs ont été satisfaits de leur interaction avec l'interface du compresseur, ils ont donné une note moyenne d'environ 4 sur 5. Le taux de compression du mode semi-automatique est nettement supérieur à celui du mode manuel, d'environ 15 %, pour une qualité des compressions équivalente. Le taux de compression du mode automatique se situe entre ceux des 2 autres modes, pour une qualité des compressions légèrement inférieure et un temps de compression non comparable, car de l'ordre de 5 secondes par texte pour 200 à 300 secondes pour les autres modes, ce qui confère au mode automatique le meilleur rapport de temps/taux/qualité de compression. Parmi les 4 types de compression automatique, celui qui conserve les modifieurs circonstanciels de lieu et de temps de la proposition, en effaçant les autres circonstants a obtenu les meilleurs résultats, et se place donc en paramétrage favori pour le mode automatique. Le genre journalistique s'est révélé être le plus propice à la compression. Le genre scientifique se compresse aussi bien que le narratif, alors que nous pensions qu'il serait peu sujet à la compression. La qualité des résumés produits est bien subjective, car dans les 3 modes de compression nous retrouvons des notes faibles attribuées pour des raisons de préférence plutôt que d'incohérence, ce qui nous conforte dans notre protocole d'évaluation, où ne sont pas utilisés des résumés

de référence pour déterminer la qualité des compressions. Nos résultats sont sensiblement dégradés à cause d'une analyse syntaxique non parfaite, malgré nos prétraitements, et de quelques bugs dans CoLin, au niveau de la phase d'assemblage des mots conservés dans les compressions.

Nos résultats sont donc très encourageants, car nous avons pu montrer la réelle aide au résumé automatique apportée par CoLin, ainsi que la qualité de ses compressions automatiques. Notre compresseur reste toutefois dépendant d'une analyse syntaxique, qui, si elle est partielle, peut dégrader les résultats.

Enfin, notre hypothèse théorique sur l'influence de la fonction syntaxique sur l'importance des constituants a été largement confortée par ces résultats.

6

Conclusion et perspectives

Sommaire

6.1 Synthèse 165
6.2 Perspectives 168

6.1 Synthèse

Le résumé automatique est un domaine de recherche étudié depuis près de 50 ans [Luhn, 1958]. De très nombreuses approches ont abordé cette problématique par un grand nombre de techniques différentes, comme nous l'avons vu au chapitre 2. La majorité se concentre sur la localisation des phrases importantes, dans le but de les extraire puis de les mettre bout à bout pour produire un résumé. Ces phrases sont conservées dans leur intégralité, alors qu'elles contiennent très souvent des éléments informatifs peu importants en leur sein. De plus, de telles approches tiennent peu compte d'éléments structurels du texte, ce qui a pour conséquence la production de résumés peu cohérents. Partant de ce constat nos objectifs ont été de proposer une approche de résumé de texte se basant sur une information structurelle du texte et permettant une analyse des éléments intra-phrastiques afin de réduire la taille des phrases. La structure syntaxique des phrases est l'information que nous avons trouvée la plus adéquate à cette problématique, car elle permet de donner des informations sur l'importance des constituants de la phrase et elle peut être extraite automatiquement à un niveau de qualité suffisant pour notre approche. Certaines approches comme [Knight & Marcu, 2002] exploitent la structure syntaxique pour réaliser des compressions de phrases, à partir d'un modèle statistique basé sur un moteur d'apprentissage. Ces approches sont contraintes par les limites intrinsèques d'un modèle et d'un corpus d'apprentissage. La généricité du corpus est inévitablement restreinte, notamment sur les constructions syntaxiques présentes et les genres textuels abordés.

Nous avons alors abordé le problème sous un angle différent, par la création d'un modèle basé sur une analyse manuelle et linguistique de l'importance des syntagmes dans la phrase, décrit au chapitre 3. Nous nous sommes appuyés sur la théorie du gouvernement et du liage de Noam Chomsky pour définir notre syntaxe, orientée vers l'importance syntaxique des constituants. Cette étude a abouti à la définition de deux classes de constituants gouvernés : les modifieurs et les compléments. Les premiers sont des éléments systématiquement facultatifs sur le plan syntaxique. L'effaçabilité des seconds dépend de la tête lexicale qui les sous-catégorise. Puis nous avons exploré, au sein des phrases, un ensemble de traits linguistiques jouant un rôle dans l'importance sémantique comme syntaxique des constituants. Nous nous sommes principalement concentré sur l'information de sous-catégorisation des compléments ainsi que sur les propriétés des fonctions lexicales vis-à-vis de l'importance des compléments. Cette étude a abouti sur l'esquisse d'un modèle computationnel de compression syntaxique des phrases. Nous nous sommes ensuite posé la question de l'influence du genre sur notre modèle de compression, qui devait être mieux déterminée lors de l'évaluation de notre compresseur automatique. Enfin, nous avons soulevé le problème de la limite de la localisation du contenu important dans la phrase, intrinsèque à notre modèle et au traitement automatique de la langue à l'heure actuelle. Cela nous a conduit à ouvrir une piste parallèle à notre méthode automatique : la production de résumé semi-automatique, par une interaction avec un utilisateur.

Ces éléments théoriques ont fait l'objet d'une mise en œuvre avec la réalisation d'un compresseur de phrases, CoLin, décrit dans le chapitre 4. Nous avons commencé par exposer l'architecture générique d'un système de compression automatique basé sur notre modèle. CoLin, construit selon cette architecture, requiert une analyse syntaxique lui fournissant les fonctions syntaxiques ainsi que d'autres informations linguistiques sur les constituants de la phrase. Cela nous a conduit à choisir un analyseur syntaxique, SYGFRAN, puis à l'intégrer à notre compresseur. Nous avons alors détaillé SYGFRAN, ainsi que SYGMART, l'environnement opérationnel qui permet d'interpréter les règles et les dictionnaires de l'analyseur. Enfin, nous avons présenté CoLin, composé d'un ensemble de règles transformationnelles interprétées par SYGMART et d'une interface Web permettant une interaction sur la production de la compression. Nos règles permettent d'adapter l'analyse de SYGFRAN à notre grammaire puis de marquer les constituants susceptibles d'être effacés et enfin de linéariser la structure arborescente en une chaîne textuelle balisée, qui est enfin transmise à l'interface Web. Cette interface met en valeur, par un surlignage coloré, les constituants effaçables, tout en indiquant par un code de couleur spécifique leur importance probable. La sélection par défaut respecte notre modèle de compression, dans la limite des informations disponibles dans l'analyse. L'utilisateur peut alors valider cette proposition de sélection, s'il souhaite obtenir une compression rapide, ou peut interagir avec l'interface pour adapter le résumé à ses préférences, s'il souhaite une compression de

meilleure qualité.

Nous avons ensuite procédé à la validation de notre modèle de compression syntaxique à travers l'évaluation de CoLin dans une expérimentation décrite au chapitre 5. Nous avons défini un protocole d'évaluation manuel en nous appuyant sur celui utilisé dans [Knight & Marcu, 2002], et en l'adaptant à nos attentes sur deux principaux points : (1) alléger l'effort cognitif demandé à l'évaluateur et (2) évaluer aussi le mode semi-automatique de notre compresseur. Nous avons développé un système informatique en ligne permettant à un ensemble d'évaluateurs de s'inscrire, de réaliser des compressions, puis de les noter. Les évaluateurs ont été répartis sur des compressions manuelles et semi-automatiques. Les premières permettent d'évaluer le temps gagné par le mode semi-automatique et aussi d'être comparée en qualité aux compressions semi-automatiques et automatiques.

Les résultats de notre expérimentation sont très encourageants. La compression interactive de CoLin permet de gagner entre 20 % et 100 % de temps, par rapport à la manuelle. Les évaluateurs ont donné une note moyenne d'environ 4 sur 5 sur leur satisfaction d'interaction avec l'interface du compresseur. Le taux de compression du mode semi-automatique est nettement supérieur à celui du mode manuel, d'environ 15 %, pour une qualité de compression équivalente. Le taux de compression du mode automatique se situe entre ceux des 2 autres modes, pour une qualité de compression légèrement inférieure et un temps de compression non comparable, car de l'ordre de 5 secondes par texte pour 200 à 300 secondes pour les autres modes, ce qui confère au mode automatique le meilleur rapport de temps/taux/qualité de compression. Parmi les 4 types de compression automatique, celui qui conserve les modifieurs circonstanciels de lieu et de temps de la proposition, en effaçant les autres circonstants a obtenu les meilleurs résultats, et se place donc en paramétrage favori pour le mode automatique. La qualité des résumés produits est bien subjective, car dans les 3 modes de compression nous retrouvons des notes faibles attribuées pour des raisons de préférence plutôt que d'incohérence, ce qui nous conforte dans notre protocole d'évaluation, où ne sont pas utilisés des résumés de référence pour déterminer la qualité des compressions. Nos résultats sont sensiblement dégradés à cause d'une analyse syntaxique non parfaite, malgré nos prétraitements, et de quelques bugs dans CoLin, au niveau de la phase d'assemblage des mots conservés dans les compressions. Cette validation fournit donc des résultats très encourageants sur CoLin et notre modèle de compression syntaxique.

6.2 Perspectives

Au cours de notre travail, nous avons dégagé plusieurs pistes à explorer, dans la continuité de ce modèle, lesquelles nous présentons dans cette section.

6.2.1 Traitement automatique de la sous-catégorisation

Comme exposé en section 3.3.1, disposer d'une information précise sur la sous-catégorisation des têtes lexicales permet une détermination plus fine de l'importance des compléments. Cette tâche ne concerne pas directement le compresseur, mais plutôt l'analyseur syntaxique. Toutefois, l'analyseur fait partie intégrante du système de compression complet, c'est pourquoi nous considérons aussi son évolution. De plus, dans notre cas, les deux sont étroitement liés, car les deux exploitent le même module TELESI, il est donc plus facile de manipuler les structures de données échangées.

Nous avons vu que les informations de sous-catégorisation utiles à notre approche sont le caractère de complément obligatoire ou facultatif, ainsi que l'ensemble des constructions syntaxiques existantes pour chaque tête. En disposer requiert un enrichissement du dictionnaire OPALE de SYGFRAN. Cela peut être réalisé grâce à des ressources lexicales informatisées comme le Le*fff* ou le lexique-grammaire. Ces ressources sont toutefois encore peu complètes, en regard de l'ensemble des informations de sous-catégorisation extractibles de la langue.

6.2.2 Apprentissage sur l'interaction

La modification de la sélection des constituants par l'utilisateur est une information qui pourrait être exploitée pour améliorer la détermination de l'importance des constituants. Cela consisterait à définir un modèle d'apprentissage basé sur un ensemble de critères syntaxiques. Ainsi pourraient être considérés, pour chaque constituant, les critères suivants :

- la fonction syntaxique (complément, modifieur) ;
- la position dans la phrase ;
- des traits sémantiques comme le type de circonstant, la nature de l'entité (animée ou non, comestible ou non...) ;
- des propriétés du constituant parent (fonction syntaxique, catégorie lexicale...) ;
- des propriétés des constituants frères (présence de certains modifieurs, déterminants...).

Ces critères seraient certainement à considérer pour chaque genre textuel. Lorsqu'une certaine configuration de ces critères coïncide avec une suppression fréquente par les utilisateurs, les constituants vérifiant la configuration verraient alors leur importance décroître,

et de même pour l'inverse.

6.2.3 La compression de phrases dans le résumé automatique

Nous venons de présenter deux perspectives qui proposent des pistes basées sur des techniques se situant à un niveau principalement : les mots et constituants. Les éléments textuels ôtés du texte à résumer sont donc de petite granularité et le taux de compression obtenu dépasse rarement les 50 %, lorsqu'il s'agit de conserver une bonne cohérence textuelle. Pour les applications du résumé automatique requérant un taux de compression supérieur, il est important de considérer un plus grand niveau de granularité d'analyse. Juste au-dessus du constituant, la phrase est l'unité textuelle la plus considérée à l'heure actuelle.

Nous proposons maintenant deux pistes qui, comme notre approche s'appuient sur une structure du texte pour déterminer l'importance de segments textuels, mais qui travaillent à un niveau de granularité pouvant atteindre ou dépasser la phrase.

Les marqueurs lexicaux des fonctions rhétoriques. Déterminer la structure rhétorique d'un texte n'est pas une tâche facilement accessible de nos jours, comme vu en section 2.3.3.3. Cependant, certains indices au niveau de la phrase peuvent se révéler être assez fiables pour déterminer la position de certaines fonctions rhétoriques utiles à notre approche. Parmi ces fonctions, nous pensons particulièrement aux reformulations et exemplifications, car elles fournissent une information supplémentaire et généralement non essentielle et peuvent donc faire l'objet d'une suppression. Pour la reformulation, les marqueurs lexicaux l'introduisant peuvent être *autrement dit*, *c'est-à-dire*, *en d'autres termes*... L'exemplification peut être détectée par la localisation de marqueurs tels *par exemple*, *à titre d'illustration*, *comme*, *ainsi*...

Ces fonctions peuvent couvrir quelques constituants, jusqu'à la phrase, et voire même plusieurs phrases. Il est alors nécessaire de correctement détecter le début et la fin du segment textuel impliqué.

Par exemple, pour les fonctions rhétoriques recouvrant un paragraphe, le marqueur de la fonction aura de fortes chances d'être en début de paragraphe. Il est alors utile de considérer la structure de surface du texte pour aider à la détection des frontières de ces fonctions.

Le thème comme critère d'importance des constituants et phrases. SYGFRAN, en plus de fournir une analyse syntaxique des phrases, permet aussi de calculer le thème d'une phrase ou d'un constituant, sous la forme d'un vecteur sémantique [Chauché *et al.*, 2003]. Chaque vecteur est composé de 873 dimensions, chacune corres-

pondant à un concept défini dans le thésaurus Larousse. Pour un segment textuel donné, d'après les mots utilisés, ainsi que la fonction syntaxique des constituants dans lesquels ils sont inclus, SYGFRAN détermine quels concepts sont activés et avec quelle intensité, pour ensuite produire un vecteur sémantique sensé représenter le thème de ce segment textuel.

Cette information thématique peut ensuite être utilisée dans le cadre d'un résumé automatique, si nous souhaitons orienter la production du résumé sur un certain thème T, comme celui du document ou du corpus analysé, ou encore un thème défini par le récepteur, selon ses préférences. Ce critère d'importance des segments textuels sera alors calculé par comparaison du thème du segment TS avec le thème T. La comparaison peut se faire par une fonction de similarité basée sur la mesure de l'angle des deux vecteurs : plus l'angle est petit, plus les thèmes sont proches. Ainsi, si le thème de T est proche du thème de TS, il est censé être plus important que s'il en est éloigné.

Afin de réduire davantage la taille des textes, la compression de phrases peut alors intervenir en parallèle à de tels résumés inter-phrastiques. Elle s'inscrit ainsi dans un processus plus général, comme une des tâches du résumé automatique.

Index

adjoint, 52
adverbe, 52
AGATE, 101
analyse en surface, 23
analyse partielle, 104
anaphore, 27
apposition, 75
arbre enrichi, 81
arbre syntagmatique, 40
argument, 22, 50
article, 74
article défini, 74
article indéfini, 74

catégorie fonctionnelle, 45
catégorie syntaxique, 45
Chomsky, 38
cible, 17
circonstant, 76
cohérence structurelle, 41
COLIN, 110
complément, 49, 78
complément obligatoire, 66
compression de phrases, 9, 39
compression structurelle, 27
compression syntaxique, 40
constituant, 40
constituant gouverné, 44
contenu informationnel, 17, 41
coréférence, 27
couleur d'importance, 124
couverture syntaxique, 104

détachement, 75
domaine, 16

effacement, 43
élément incident, 75
entités, 26
événements, 18, 21, 29
expression figée, 73
extraction, 22

FL, 69
fonction lexicale, 69
fréquence des termes, 24

GB, 38
genre, 16, 83
gouvernement, 38, 44
gouverneur, 44
grammaire élémentaire SYGMART, 95
grammaire de dépendance, 38
grammaires universelles, 62
granularité, 19
groupe, 40

incise, 75
inclusion des constituants, 125
interaction, 85
interaction de COLIN, 123
interface Web, 121
interrogation, 77

langue, 16
Le*fff*, 63, 64
liage, 38

médium, 15
marqueurs lexicaux, 25
modèle computationnel, 77
modificateur, 56
modifier, 56
modifieur, 56, 78

négation, 76
nature des constituants, 26

OPALE, 91

période spaciale, 18
période temporelle, 18
patrons, 21
phrasème complet, 73
position dans la phrase, 76
position dans le texte, 25
prédicat, 22
processus de production, 20
profondeur d'analyse, 20
pronom, 54
proposition incise, 75

récepteur, 13
réseau de grammaires SYGMART, 94
résumé, 13
résumé automatique, 7
résumé par compression de phrases, 41
résumé par extraction, 22
résumé par reformulation, 21
rôles thématiques, 31
règle transformationnelle SYGMART, 95
règles de compression, 113
règles de transformation, 45
règles structurelles, 45, 58
reformulation, 21
rhétorique, 30
RST, 30

sélection de constituants, 124
sélection des constituants, 87
saillance, 18
similarité lexicale, 26
source, 15
sous-catégorisation, 62
sous-catégorisation partielle, 68
soustraction, 43
spécifieur, 48
structure événementielle, 29
structure des rôles thématiques, 31
structure rhétorique, 30
structure syntaxique, 32, 40
structure textuelle, 27
structure thématique, 29
style, 18
suppression, 43
SYGFRAN, 91, 101
SYGMART, 91
syntagme, 40
syntaxe ambiguë, 104
synthèse, 15
système de règles, 58

tête fonctionnelle, 45
tête syntaxique, 44
taille, 18
TALN, 7, 27
TELESI, 93
théorie du gouvernement et du liage, 38
thème, 17, 29

unité textuelle, 19
unités textuelles, 19
UT, 19

vocabulaire, 16

X-barre, 45

A

Glossaire

A.1 Sigles et acronymes

BE : *Basic Elements*

CoLin : COmpresseur LINguistique

DUC : *Document Understanding Conferences*

FL : Fonction Lexicale

FS : Fonction Syntaxique

GB : *theory of Government and Binding*

GC : Grammaire Classique

GPHC : Grammaire Probabiliste Hors Contexte

K&M : approche de Knight et Marcu, [Knight & Marcu, 2002]

MCP : Modifieurs Circonstanciels de la Proposition

ROUGE : *Recall-Oriented Understudy for Gisting Evaluation*

RST : *Rhetorical Structure Theory*

SC : Sous-Catégorisation

SR : Structure Rhétorique

Sygmart : Système Grammatical de Manipulation Algorithmique et Récursive de Texte

TALN : Traitement Automatique des Langues Naturelles

Telesi : Transduction d'ÉLÉment Structurés Indexés

TLFi : Trésor de la Langue Française informatisé

UT : Unité textuelle

A.2 Catégories des têtes lexicales et fonctionnelles de l'approche théorique

A : Adjectif
Adv : Adverbe
Conj : Conjonction
D : Déterminant
I : tête fonctionnelle de flexion du verbe (*Inflection*)
N : Nom
P : Préposition
V : Verbe
X^0 : Tête syntaxique ou fonctionnelle
XP (NP, VP, IP...) : Constituant gouverné par une tête de catégorie lexicale X

A.3 Variables de SYGFRAN et COLIN

CAT : catégorie grammaticale
DEBCONST : position du premier caractère du premier mot d'un constituant
DECDEBCONST, DECFINCONST : niveau d'imbrication d'un constituant par rapport à ses descendants
FINCONST : position du premier caractère du dernier mot d'un constituant
FRM : chaîne d'entrée (mot du texte sous sa forme originale)
K : catégorie du groupe (catégorie grammaticale de la tête lexicale)
LEMME : chaîne textuelle du lemme
PLACEMOT : position du premier caractère d'un mot
SEMOBJ : sémantique de l'objet (type de circonstant)
SUPPRTYPE : type de suppression d'un constituant
VERROU : protection d'un constituant contre l'effacement

A.4 Valeurs de SYGFRAN et COLIN

ADJ : adjectif
COMP : complément
DET : déterminant
GADV : groupe adverbial
GA : groupe adjectival
GN : groupe nominal

GV : groupe verbal
GNPREP : groupe nominal prépositionnel
LIEU : circonstant de lieu
MOD : modifieur
N : nom
PH ou PHRASE : phrase ou proposition
PHAMBG : construction syntaxique ambiguë (pour la phrase)
PREP : préposition
REP : représentant
TEMPS : circonstant de temps
ULFRA : unité linguistique française inconnue
ULOCC : locution inconnue (*Unknown Locution*)
V : verbe

B

Extraits des corpus exploités au cours de la thèse

B.1 Conte polynésien correctement analysé par SYGFRAN

MAUI PART A LA RECHERCHE DE SES PARENTS.

A partir de ce soir-là, Maui fut le favori de sa mère : même s'il faisait des bêtises, elle ne le grondait pas. Quand ses frères protestaient, il se moquait d'eux parce qu'il savait avoir la protection de sa mère. Mais pendant son absence, il devait faire attention à ne pas dépasser les limites, sinon il risquait d'être puni par eux au cours de la journée.

Une nuit, Maui imagina un tour à jouer à sa mère afin de découvrir où elle allait. Une fois tous les autres endormis sur leurs nattes, il se releva et fit le tour de la maison, examinant les grands stores tressés qui la fermaient pour la nuit. Partout où filtrait la clarté d'une étoile, il bouchait vite l'ouverture avec des étoffes d'écorce et calfeutrait même les fentes avec des roseaux. Puis il déroba le manteau, la ceinture et la couronne de sa mère et les cacha en se disant qu'il en aurait besoin plus tard.

Maui reprit alors sa place sur les nattes et décida de rester éveillé. La longue nuit passa lentement sans que sa mère ne bouge. Quand vint le matin, pas un rai de lumière ne put percer pour éveiller les dormeurs. Bientôt ce fut l'heure où le soleil grimpait au-dessus de l'horizon. D'habitude Maui pouvait distinguer dans la pénombre les formes des pieds de ses frères à l'autre bout de la maison, mais ce matin il faisait trop noir. Et sa mère continuait à dormir.

Au bout d'un moment elle bougea et marmonna : "Quelle sorte de nuit est-ce donc pour durer si longtemps ?" Mais elle se rendormit parce qu'il faisait aussi noir qu'au coeur de la nuit dans la maison.

Finalement elle se réveilla en sursaut et se mit à chercher ses vêtements. Courant de

tous côtés, elle arracha ce que Maui avait fourré dans les fentes. Mais c'était le jour ! Le grand jour ! Le soleil était déjà haut dans le ciel ! Elle s'empara d'un morceau de tapa pour se couvrir et se sauva de la maison, en pleurant à la pensée d'avoir été ainsi trompée par ses propres enfants.

Sa mère partie, Maui bondit près du store qui se balançait encore de son passage et regarda par l'ouverture. Il vit qu'elle était déjà loin, sur la première pente de la montagne. Puis elle s'arrêta, saisit à pleines mains un arbuste de tiare Tahiti, le souleva d'un coup : un trou apparut, elle s'y engouffra et remit le buisson en place comme avant.

Maui jaillit de la maison aussi vite qu'il put, escalada la pente abrupte, trébuchant et tombant sur les mains car il gardait les yeux fixés sur l'arbuste de tiare. Il l'atteignit finalement, le souleva et découvrit une belle caverne spacieuse qui s'enfonçait dans la montagne.

Plus tard, tandis que ses frères étaient très affairés à se baigner dans le frais ruisseau et à chercher un fruit de l'arbre à pain à se mettre sous la dent, Maui les questionna : "Où croyez-vous donc que notre père et notre mère passent la journée ?"

-"Comment le saurions-nous ? Et pourquoi te tracasses-tu avec ça ? Tu ne peux donc pas vivre tranquillement avec nous ? Que nous importe notre père ou notre mère ? Est-ce qu'elle nous a élevés avec de la bonne nourriture ? Pas du tout : elle était toujours partie. Peut-être bien que le grand Ta'aroa, dieu du ciel, est notre père et qu'il a envoyé ses enfants ici-bas pour s'occuper de nous ! Niu-Hiti, la douce brise qui rafraîchit la terre et les jeunes plantes ; Hau-Ri'i, le vent humide qui les mouille ; Hau-Roto-Roto, le beau temps qui les fait pousser ; Tou-Ari'i, le dieu de la pluie qui les arrose, et son frère, Ti-Ari'i, qui les nourrit de ses rosées. Ta'aroa a envoyé toute sa famille pour permettre à notre nourriture de pousser. Ensuite Papa, la grande déesse mère de la terre, a fait germer ses graines pour nous tous qui sommes ses enfants."

-"Mais oui, bien sûr", leur répondit le petit Maui, "tout ce que vous dites est vrai. C'est même encore plus vrai pour moi que pour vous, parce que c'est la mer qui a été ma nourrice et ses bouillonnements d'écume mon lait. Vous, vous avez été nourris au lait de notre mère avant de pouvoir manger d'autres nourritures. Mais moi, ô mes frères, je n'ai jamais tété son sein, ni rien mangé de sa main. Et pourtant je l'aime pour l'unique raison qu'elle m'a porté en elle, et c'est parce que je l'aime que je souffre de ne pas savoir où elle se trouve avec mon père."

Ses frères se sentirent surpris et charmés par le petit Maui quand ils l'entendirent parler de cette façon. Après avoir réfléchi un moment à ce qu'il avait dit, ils l'approuvèrent et l'encouragèrent à tenter de trouver leur père et leur mère.

Maui ne se tint plus de joie. Il se mit tout de suite à faire la magie dont il savait avoir besoin pour pénétrer dans la caverne sous l'arbuste de tiare et trouver son chemin souterrain dans l'autre monde. Il allait devoir voyager vite et il décida de se changer

en oiseau. Il ne savait pas quel oiseau choisir. Il pensa bien sûr au noha, le pétrel qui niche dans un terrier de la montagne, mais il le jugea trop gros. Il se fit maho, marouette fuligineuse, mais ses frères pensèrent qu'il était trop petit et pas joli. Puis il devint otaha, grande frégate noire, mais ils trouvèrent effrayante cette créature aux ailes plus longues que leurs bras. Alors il essaya un oiseau après l'autre, le 'uriri, petit chevalier voyageur à la voix claire et hardie, le tarapapa, hirondelle de mer bruyante à la voix gutturale, le kivi, courlis chasseur des petits crabes rouges, au long bec courbé comme un manche d'outil, le otu'u, aigrette sacrée qui tangue sur ses hautes échasses, le torea, pluvier doré, puis le a'o, fou brun, et le ua'ao, fou à pieds rouges, trop comiques, et puis le itata'e tout blanc et le oa tout brun, jusqu'à ce qu'il ait pris l'apparence de tous les oiseaux du monde, tour à tour. Enfin il se changea en pigeon vert.

B.2 Extraits du corpus d'évaluation

B.2.1 Premier document narratif, *Vingt mille lieues sous les mers* de Jules Verne, extrait du premier chapitre

L'année 1866 fut marquée par un événement bizarre, un phénomène inexpliqué et inexplicable que personne n'a sans doute oublié. Sans parler des rumeurs qui agitaient les populations des ports et surexcitaient l'esprit public à l'intérieur des continents, les gens de mer furent particulièrement émus. Les négociants, armateurs, capitaines de navires, skippers et masters de l'Europe et de l'Amérique, officiers des marines militaires de tous pays, et, après eux, les gouvernements des divers États des deux continents, se préoccupèrent de ce fait au plus haut point.

En effet, depuis quelque temps, plusieurs navires s'étaient rencontrés sur mer avec « une chose énorme » un objet long, fusiforme, parfois phosphorescent, infiniment plus vaste et plus rapide qu'une baleine.

Les faits relatifs à cette apparition, consignés aux divers livres de bord, s'accordaient assez exactement sur la structure de l'objet ou de l'être en question, la vitesse inouïe de ses mouvements, la puissance surprenante de sa locomotion, la vie particulière dont il semblait doué. Si c'était un cétacé, il surpassait en volume tous ceux que la science avait classés jusqu'alors. Ni Cuvier, ni Lacépède, ni M. Dumeril, ni M. de Quatrefages n'eussent admis l'existence d'un tel monstre - à moins de l'avoir vu, ce qui s'appelle vu de leurs propres yeux de savants.

A prendre la moyenne des observations faites à diverses reprises - en rejetant les évaluations timides qui assignaient à cet objet une longueur de deux cents pieds et en repoussant les opinions exagérées qui le disaient large d'un mille et long de trois - on pouvait affirmer, cependant, que cet être phénoménal dépassait de beaucoup toutes les dimensions admises jusqu'à ce jour par les ichtyologistes - s'il existait toutefois.

Or, il existait, le fait en lui-même n'était plus niable, et, avec ce penchant qui pousse au merveilleux la cervelle humaine, on comprendra l'émotion produite dans le monde entier par cette surnaturelle apparition. Quant à la rejeter au rang des fables, il fallait y renoncer.

B.2.2 Premier document scientifique, extrait du corpus de la conférence DEFT'06

Si l'on possède une description des données par un ensemble de D attributs, le problème de la sélection d'attributs consiste à chercher un sous-ensemble de d attributs qui préserve au mieux les informations nécessaires à l'algorithme d'apprentissage. Cette technique sera de nouveau évoquée un peu plus loin au paragraphe [REFERENCE] à l'occasion de la distinction entre filter methods et wrapper methods.

Au fond, on est à peu près dans la même situation que celle du réglage des paramètres d'un algorithme (voir le paragraphe [REFERENCE]) : si l'on considère l'algorithme d'apprentissage comme paramétré par le sous-espace de représentation choisi, la question est de trouver le meilleur compromis entre la complexité, mesurée ici par la valeur de d, et l'efficacité, qui est la performance de l'algorithme dans l'espace de dimension réduite de D à d.

Il y a deux difficultés au problème de la sélection d'attributs : La première est qu'en général on recherche une méthode indépendante de tout algorithme, ceci pour ne pas faire dépendre la représentation des connaissances des choix opérationnels qui suivront. Ce n'est pas toujours le cas : on peut parfois être fixé sur le choix d'un algorithme et essayer de simplifier les données sans nuire à ses performances. Mais en principe on doit trouver une façon générique de mesurer la qualité d'un sous-ensemble d'attributs par un critère J. Ce n'est pas un problème évident. Dans un problème de classification, diverses mesures absolues de séparabilité des classes ont ainsi été définies par de nombreux auteurs ([CITATION]).

La seconde difficulté est qu'il y a un grand nombre de sous-ensembles d'attributs de dimension donnée d et au total 2^D. Il est hors de question de mesurer sur chacun un critère de séparabilité ou la mesure de performance d'un algorithme particulier. On pourrait penser que la structure particulière de cet espace (l'ensemble des sous-ensembles d'un ensemble fini) permet d'utiliser des méthodes approximatives efficaces, mais il faut être prudent à ce sujet, comme le montre l'exemple qui suit.

Considérons le problème d'apprentissage de règle de classification sur un ensemble de cinq points en dimension D=3 donné à la figure [REFERENCE]. Il est facile de voir que les deux classes (représentées par les symboles et o) sont bien séparées, au moins sur cet ensemble d'apprentissage.

B.2.3 Premier document journalistique, extrait d'un article publié sur le site internet le 27 février 2007, intitulé « Darfour : la Cour pénale internationale désigne les criminels de guerre »

Le procureur de la Cour pénale internationale (CPI), Luis Moreno Ocampo, a demandé aux juges d'assigner ou d'émettre des mandats d'arrêt à l'encontre de deux hauts responsables des crimes commis au Darfour. Selon le parquet, Ahmad Muhammad Harun, ancien ministre de l'intérieur du gouvernement soudanais et Ali Muhammad Ali Abd-Al-Rahman, l'un des commandants des milices "janjawids" (cavaliers armés), alliées aux forces gouvernementales dans la guerre qui oppose depuis quatre ans Khartoum aux mouvements rebelles, auraient commis des crimes contre l'humanité et des crimes de guerre au Darfour en 2003 et 2004. Dans un document remis aux magistrats mardi matin 27 février, le procureur a relevé 51 charges contre les deux hommes et fait état de meurtres, d'exécutions sommaires, de pillages, de viols et de déplacements forcés de populations.

Premier visé, Ahmad Muhammad Harun, ex-ministre de l'intérieur du gouvernement soudanais, placé à la tête du Bureau sécurité du Darfour, aurait armé, financé et placé les hommes à la tête des janjawids, sachant qu'ils "combattaient aux côtés des forces gouvernementales" dans une guerre qui, depuis 2003, aurait fait au moins 250 000 morts et 2,5 millions de déplacés.

Selon le document remis par le parquet aux juges, Ahmad Harun s'est rendu toutes les trois semaines au Darfour au cours de l'année 2003, par avion depuis Khartoum, pour payer les miliciens. Il aurait été vu dans un avion transportant des fusils d'assaut G3s et kalachnikov. Or l'ancien ministre savait que les janjawids " attaquaient les populations civiles et commettaient des crimes ", écrit le procureur. Ses fonctions lui permettaient d'accéder aux informations fournies par l'armée, la police et les services de renseignement. D'après plusieurs témoins, il aurait notamment déclaré avoir " le pouvoir et l'autorité de tuer et de pardonner ".

En 2003, Ali Muhammad Ali Abd-Al-Rahman, dit "Ali Kushayb", chef janjawid, commandait, selon le parquet, des "dizaines de milliers" de miliciens, et "a personnellement dirigé" plusieurs attaques dans l'ouest de la province soudanaise. Ces attaques ne visaient pas spécifiquement les mouvements rebelles, estime le parquet, mais étaient dirigées contre des villageois accusés de les soutenir. Cette stratégie est devenue la justification pour les meurtres de masse, les exécutions sommaires, et les viols de civils qui ne participaient pas au conflit armé. La même stratégie a entraîné le déplacement forcé de villages entiers.

C

Règles SYGMART

Code source C.1 – Extrait de la grammaire TELESI de CoLin de post-traitement à SYGFRAN pour la correction de constructions syntaxiques.

```
&REFER(VarCompress,GramPostProcess).

&GRAMMAIRE.

&ENTREE:POSTPROCESS.

// [...]

--> CONSTR_SYNT_NAP.

// ************** CONSTRUCTIONS SYNTAXIQUES DES NOMS, ADJECTIFS ET PRÉPOSITIONS ***************

&GRAM: CONSTR_SYNT_NAP(I).
CSN_EXTERNE_A: 0(1(2),3(*,%4,*,5)) /
   0:(K=GN)|(K=GNPREP); 1:(K=GA); 2:(LEMME='externe'); 3:(K=GNPREP); 4:(CAT=PONCT); 5:(LEMME='à')
   => 0(x(*1<,2>*,2,*1<2,>*,3(%4,5))) / x:1; 3:3(FS=ATTR; TYP=ADADJ).
CSN_LA_JUSTIFICATION_POUR: 0(1(2(*,3,*,4,*)),5(*,%6,*,7)) /
   0:(K=PHRASE); 1:(K=GV); 2:(K=GN); 3:(SOUSD=ARTD); 4:(LEMME='justification');
   5:(K=GNPREP); 6:(CAT=PONCT); 7:(LEMME='pour')
   => 0(1(2(3,4,5(%6,7)))) / 5:5(FS=ATTR).
CSN_DE_PLUS_EN_PLUS_DE_QQUECHOSE: 0(*,1,*),*,2(3) /
   0:(K=GADV);1:(LEMME='de plus en plus de'); 2:(K=GN); 3:(LEMME='initiative')
   => x(1,*2<,3>*,3,*2<3,>*) / x:2; 1:1(CAT=DETERM).

// ************** CONSTRUCTIONS SYNTAXIQUES DES VERBES ***************

&GRAM: CONSTR_SYNT_VERBES(I).
CSV_ACCEDER_A: 0(1(2),3(*,%4,*,5)) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='accéder'); 3:(K=GNPREP); 4:(CAT=PONCT); 5:(LEMME='à')
   => 0(x(*1<,2>*,2,*1<2,>*,3(%4,5))) / x:1; 3:3(FS=OBJI).
CSV_AGIR_COMME: 0(1(2),3(*,%4,*,5)) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='agir'); 3:(KPH=PHCONJ); 4:(CAT=PONCT); 5:(LEMME='comme')
   => 0(x(*1<,2>*,2,*1<2,>*,3(%4,5))) / x:1; 3:3(FS=OBJI).
CSV_CONSOMMER_DE: 0(1(2),3(*,%4,*,5)) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='consommer'); 3:(K=GNPREP); 4:(CAT=PONCT); 5:(LEMME='de')
   => 0(x(*1<,2>*,2,*1<2,>*,3(%4,5))) / x:1; 3:3(FS=OBJI).
```

```
CSV_ENTRAINER_QQUEPART: 0(1(2),3) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='entraîner'); 3:(K=GNPREP)&(SEMOBJ=LIEU)
   => 0(x(*1<,2>*,2,*1<2,>*,3)) / x:1; 3:3(FS=OBJI).
CSV_EN_VENIR_A: 0(1(2(3),*,4),5(*,%6,*,7)) /
   0:(K=PHRASE); 1:(K=GV); 2:(K=GADV); 3:(LEMME='en'); 4:(LEMME='venir');
   5:(KPH=PHINFPREP); 6:(CAT=PONCT); 7:(LEMME='à')
   => 0(x(*1<,2>*,2(3),4,*1<2,>*,5(%6,7))) / x:1; 5:5(FS=OBJI).
CSV_ENVOYER_QQUECHOSE_QQUEPART: 0(1(2,3),4) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='envoyer'); 3:(FS=OBJT); 4:(K=GNPREP)&(SEMOBJ=LIEU)
   => 0(x(*1<,2>*,2,*1<2,3>*,3,*1<3,>*,4)) / x:1; 4:4(FS=OBJI).
CSV_ETRE_DIRIGE_CONTRE: 0(1(2,3(4)),5(*,%6,*,7)) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='être'); 3:(K=GA); 4:(LEMME='dirigé');
   5:(K=GNPREP); 6:(CAT=PONCT); 7:(LEMME='contre')
   => 0(x(*1<,2>*,2,*1<2,3>*,3(4),*1<3,>*,5(%6,7))) / x:1; 5:5(FS=OBJI).
CSV_FOURRER_QQUECHOSE_DANS_QQUECHOSE: 0(1(2),3) /
   0:(K=PHRASE); 1:(K=GV)&(FOBJ=1); 2:(LEMME='fourrer'); 3:(K=GNPREP)&(SEMOBJ=LIEU)
   => 0(x(*1<,2>*,2,*1<2,>*,3)) / x:1; 3:3(FS=OBJI).
CSV_ORDONNER_QUE: 0(1(2),3) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='ordonner'); 3:(KPH=PHCONJ)
   => 0(x(*1<,2>*,2,*1<2,>*,3)) / x:1; 3:3(FS=OBJT).
CSV_S_ALARMER_DE: 0(1(2(3),*,4),5(*,%6,*,7)) /
   0:(K=PHRASE); 1:(K=GV); 3:(CAT=REP)&(SOUSR $>= REFL); 4:(LEMME='alarmer');
   5:(K=GNPREP)|(KPH=PHINFPREP); 6:(CAT=PONCT); 7:(LEMME='de')
   => 0(x(*1<,2>*,2(3),4,*1<2,>*,5(%6,7))) / x:1; 5:5(FS=OBJI).
CSV_TENIR_POUR: 0(1(2),3(*,%4,*,5)) /
   0:(K=PHRASE); 1:(K=GV); 2:(LEMME='tenir'); 3:(K=GNPREP); 4:(CAT=PONCT); 5:(LEMME='pour')
   => 0(x(*1<,2>*,2,*1<2,>*,3(%4,5))) / x:1; 3:3(FS=OBJI).

// [...]
```

Code source C.2 – Extrait de la grammaire TELESI de COLIN de correction *ad hoc* des cas d'analyses partielles des phrases du corpus d'évaluation, ici pour le corpus journalistique.

```
&REFER(VarCompress,GramEvalAdHoc).

&GRAMMAIRE.

&ENTREE: EVALADHOC.
--> JOUR1.

&GRAM:JOUR1(E).
JOUR1_4:*(1(2(*,3(*,4(*,5,*),*,6,*,7(*,8,*,9,*,10,*,11,*,12,*),*)))) / 9:(FRM='Ahmad')
   => 1(2(3(A(4(5),6,8),9,10,11,12))) / A:(K=GN; FS=ATTR; LEMME='GN').
JOUR1_5:*(1(2(*,3,*,18,*,21(*,22,*,24,*,25,*),*,26,*,32(*,33,*,34,*,35,*)))) / 35:(FRM='Darfour')
   => 1(2(3,18,21(22,24,25,32(33,34,35)),26)) / 32:32(FS=OBJT).
JOUR1_7:*(1(2(*,3,*,35(*,36,*,37,*),*,40,*))) / 3:(KPH=PHCONJ); 36:(FRM='écrit')
   => 1(2(35(37,36,3),40)) / 3:3(FS=OBJT); 37:37(FS=SUJ).
JOUR1_9:*(1(2(*,3(*,4,*,11,*,13(*,14,*,15,*,16,*),*),*,18(*,19(*,20,*),*),*,21,*,
   22,*,25(*,26,*),*,27,*,30,*,40))) / 2:(FLX='ULFRA'); 15:(FRM='déclaré')
   => 1(*1<,2>*,3(4,11,13(14,15,16,18(19(20,A(21,B(22,26,27,30),40))))),*1<2,>*) /
   3:3(POSITION=SOMMET_PHRASE); A:(FS=OBJT; LEMME='GN'); B:(K=GN; LEMME='GN'); 30:30(FS=ATTR).
--> JOUR2.

&GRAM:JOUR2(E).
JOUR2_2:*(1(2(*,3,*,35(*,36,*,37,*),*,38(*,39,*,40,*,41,*),*,44))) / 37:(FRM='Sarkozy')
   => 1(2(3,35(36,37),38(39,40,41,44))) / 44:44(FS=OBJT).
```

```
JOUR2_8:*(1(2(*,3,*,4,*,6(*,7,*,8,*),*,9,*,27))) / 3:(FRM='"'); 8:(FRM='choisi'); 27:(FRM='"')
   => 1(2(3,4,6(7,8,9),27)) / 9:9(FS=OBJT).
JOUR2_20:*(1(2(*,3(*,4,*),*,5,*,6,*,8(*,9,*),*,10))) / 4:(FRM='2'); 9:(FRM='replacer')
   => 1(2(3(4),5,6,8(9),10)) / 10:10(FS=OBJT).
--> JOUR3.

&GRAM:JOUR3(E).
JOUR3_1:*(1(2(*,3(*,4,*,5,*,6,*,12,*),*,17,*,20))) / 5:(FRM='chef'); 20:(CASPREPSIMPLE=DEVANT)
   => 1(2(3(4,5,6,12),17,20)) / 20:20(FS=OBJT).
JOUR3_10:*(1(2(*,3(*,4(*,5,*,6),*,16(*,17,*),*),*,18,*,23(26(*,27,*,29(*,30,*),35,*),*),*,45,*))) /
   2:(FLX='ULFRA'); 6:(FRM='pénuries')
   => 1(*1<,2>*,3(4(5,6),16(17,23(26(27,29(30,35)))),18,45),*1<2,>*) /
   3:3(POSITION=SOMMET_PHRASE); 18:18(FS=COMPCIR); 23:23(FS=OBJT); 35:35(FS=OBJT).
--> JOUR4.

&GRAM:JOUR4(E).
JOUR4_3:*(1(2(*,3(*,4,*,9,*,19(*,20,*,21,*),*,28,*),*,34(39,*),*,41,*,
   42(57(59(*,60,*,61,*,62,*),*),*),*,70,*,71,*,74,*,75,*,76,*))) /
   2:(FLX='ULFRA'); 61:(FRM='réticulocytes')
   => 1(*1<,2>*,3(4,9,19(20,21,34(39,42(41,57(59(60,61,62,70,71)),74,75))),76),*1<2,>*) /
   3:3(POSITION=SOMMET_PHRASE); 34:34(FS=OBJT); 42:42(FS=ATTR).
JOUR4_6:*(1(2(*,3,*,4(*,5,*,9,*,10(*,11,*,13,*,15(*,16(29,*),*,33,*,34(55,*))),*,82,*,
   83(85(87,*),*),*,88,*,89,*,97(*,98,*),*,99,*,128,*),*,129,*,162,*))) /
   4:(FLX='ULFRA'); 87:(FRM='Séralini')
   => 1(2(3,10(5,9,11,13,15(16(29),33,34(55)),82,83(85(87),A(88,89,98,99)),128),129,162)) /
      A:(LEMME='GN'; K=GN; FS=ATTR); 99:99(K=GN; LEMME='GN');
      15:15(FS=OBJT); 29:29(FS=COMPCIR); 55:55(FS=COMPCIR); 83:83(K=GV;LEMME='GV'); 85:85(FS=SUJ).
JOUR4_7:*(1(2(*,3,*,10(13(16(*,17,*,18(27(34(38(43,*),*),*),*),*),*),*),*,48,*))) / 17:(FRM='tirées')
   => 1(2(3,10(13(16(17,18(27(34(38),43))))),48)) / 34:34(FS=OBJT); 43:43(FS=COMPCIR).
JOUR4_10:*(1(2(*,3(*,4,*,5,*,6,*),*,9(17(24,*),*),*,40,*))) / 5:(FRM='chercheurs')
   => 1(2(3(4,5,6),9(17(24)),40)) / 24:24(FS=OBJT).
--> JOUR5.

&GRAM:JOUR5(E).
JOUR5_12:*(1(2(*,3,*,5(*,6,*,7,*),*,57(60,*),*,61,*))) / 60:(FRM='contexte')
   => 1(2(3,5(6,57(60),7),61)) / 57:57(FS=ATTR; SOUSATTR=ATTRSUJ).
JOUR5_13:*(1(2(*,3(*,4,*,9(*,10,*,11(*,12,*,13,*,15,*),*),*),*,21,
   *,48,*,49,*,51(*,52,*),*,53(*,54,*),*,65,*))) /
   2:(FLX='ULFRA'); 51:(FLX='ULOCC'); 52:(FRM='en')
   => 1(*1<,2>*,3(4,9(10,11(12,13,15,21),53(48,49,52,54)),65),*1<2,>*) /
      53:53(FS=OBJT); 21:21(FS=OBJT).
--> NARR1.

// [...]
```

Code source C.3 – Grammaire TELESI de COLIN de marquage des anaphores probables.

```
&REFER(VarCompress,GramAnaphores).

&PROC:AFCT.
// type d'anaphore égal 1 + (1 si le nombre commun est singulier) + (2 si le genre commun est masculin)
SET_TYPE_ANA(x,y):
   TMP(FREG1) = 1;
   NUM(FREG1) = NUM(x) & NUM(y);
   GNR(FREG1) = GNR(x) & GNR(y);
   <NUMREF(y) != 0: NUM(FREG1) = %(NUM)<-NUMREF(y)>;
```

```
   <NUM(FREG1) != 0:<NUM(FREG1) = SIN: TMP(FREG1) = TMP(FREG1) + 1>>;
   <GNR(FREG1) != 0:<GNR(FREG1) = MAS: TMP(FREG1) = TMP(FREG1) + 2>>.

&PROC: CONDITION.
// il y a une anaphore potentielle entre un GN et un pronom
   ANA_POT(x,y):
      (  ((GNR(x) & GNR(y) != 0)&(NUM(x) & NUM(y) != 0)&(CAT(y) = REP)&(SOUSR(y) != REFL)&(K(y) = GN))|
         ((NUM(x) & %(NUM)<-NUMREF(y) != 0)&(NUMREF(y) != 0)&
            ((SEM(x) = PERSONNE)|(FS(x) = SUJ))&(PERSREF(y) = 3))
      )&
      (PLACEMOT(x) != PLACEMOT(y)).

&GRAMMAIRE.

&ENTREE: ANAPHORE.
--> MARQUEPREMPH.

// détermine l'ordre de sélection des couples de phrase
&GRAM: MARQUEPREMPH(U).
MARQUEPREMPH: *(0(*,1))
   => 0(1) / 1:1(TMP1 = 1).
--> ANA_2_PH.

// sélectionne 2 phrases
&GRAM: ANA_2_PH(I).
RISOL(@APPEL_ANA_2_PH_REC; x): *(0(1,*,2)) / 1:(TMP1 = 1)
   => x(*0<,1>*,C(1,2),*0<2,>*) / C:(TMP2 = 1); 1:1(TMP1 = 0); 2:2(TMP1 = 1).
--> DERNIERAPPEL.

&GRAM: DERNIERAPPEL(U).
DERNIERAPPEL(@APPEL_ANA_2_PH_REC; x): *(0(1,*))
   => x(*0<,1>*,C(1,B)) / C:(TMP2 = 1); 1:1(TMP1 = 0); B:(TMP1 = 1).
-->CLEAN.

// continue le traitement sur ces 2 phrases uniquement
&GRAM: APPEL_ANA_2_PH_REC(U).
RAPPEL(@ANAPHORE_PH; 1): *(0(1)) / 1:(TMP2 = 1)
   => 0(1).
--> RELIMINE.

// remonte sous 0 les fils de 1, puis retour à ANA_2_PH
&GRAM: RELIMINE(U).
RELIMINE: *(0(1)) / 1:(TMP2 = 1)
   => x(*0<,1>*,*1*,*0<1,>*).
--> %STOP.

&GRAM: ANAPHORE_PH.
--> MARQUE_TYPES.

// marque les types de GN et pronoms
&GRAM: MARQUE_TYPES(I).
MARQUE_TYPES: *(0?(1,2)) /
   1:((K = GN)|(K = GNPREP))&(AREP = 0)&(CAT != REP);
   2:(PERS != 1)&(PERS != 2) /
      ANA_POT(1,2)&((CAND_ANA_GN(1) = 0)|(CAND_ANA_PRON(2) = 0))
         => 0(1,2) /
            1:1(SET_TYPE_ANA(1,2); CAND_ANA_GN = TMP(FREG1));
```

```
            2:2(CAND_ANA_PRON = TMP(FREG1)).
--> MARQUE_GN_GAUCHE.

// ne traite que les GN de la phrase de gauche
&GRAM: MARQUE_GN_GAUCHE(I).
MARQUE_GN_GAUCHE: *(0(*,1?(2))) / 1:(FLX!='ULFRA'); 2:(TMP3 = 0)&(CAND_ANA_GN != 0)
   => 0(1(2)) / 2:2(TMP3 = 1).
--> SUPPR_MAUVAIS_CDDGN.

// ne conserve que les pères si les fils sont même genre et nombre
&GRAM: SUPPR_MAUVAIS_CDDGN(I).
SUPPR_MAUVAIS_CDDGN: *(0?(1?(2))) / 1:(CAND_ANA_GN != 0) / CAND_ANA_GN(1) = CAND_ANA_GN(2)
   => 0(1(2)) / 2:2(CAND_ANA_GN = 0).
--> ANA_REC.

&GRAM: ANA_REC(I).
// identifie un pronom candidat
APPEL_REC(@ASSOCIE_GN; 0): *(0?(1)) / 1:(CAND_ANA_PRON != 0)
   => 0(1) / 1:1(TRAITE_ANA = 1; TMP(FREG1) = CAND_ANA_PRON).
--> %STOP.

// associe les GN au pronom choisi
&GRAM: ASSOCIE_GN(I).
ASSOCIE_GN: *(0?(1,2)) / 1:(CAND_ANA_GN = TMP(FREG1))&(AREP = 0)&(TMP3 = 1); 2:(TRAITE_ANA = 1)
   => 0(1,2) / 1:1(AREP = OUI; POSREP = PLACEMOT(2)).
--> SUPPR_GN.

&GRAM: SUPPR_GN(I).
// un GN puis un pronom non sélectionné, alors le GN n'est pas associé au bon, on verra plus tard pour lui
SUPPR_GN1: *(0?(1,2,3)) / 1:(AREP = OUI);
   2:(TRAITE_ANA = 0)&(CAND_ANA_PRON = TMP(FREG1)); 3:(TRAITE_ANA = 1)
   => 0(1,2,3) / 1:1(AREP = 0; POSREP = 0).
// 2 GN consécutifs avant le pronom sélectionné, donc le 1er n'est pas le bon
SUPPR_GN2: *(0?(1,2,3)) / 1:(AREP = OUI);
   2:(CAND_ANA_GN = TMP(FREG1)); 3:(TRAITE_ANA = 1)
   => 0(1,2,3) / 1:1(AREP = 0; POSREP = 0).
--> CLEAR_PRON.

// on passe au pronom suivant
&GRAM: CLEAR_PRON(U).
CLEAR_PRON: 0 / 0:(TRAITE_ANA = 1)
   => 0 / 0:0(TRAITE_ANA = 0; CAND_ANA_PRON = 0).
--> %STOP.

&GRAM: CLEAN(I).
CLEAN0: *(0(1,*)) / 1:(TMP1 = 1) => 0.
CLEAN1: 0 / 0:(CAND_ANA_GN != 0) => 0 / 0:0(CAND_ANA_GN = 0).
CLEAN3: 0 / 0:(TMP3 = 1) => 0 / 0:0(TMP3 = 0).
--> %STOP.

&FIN.
```

Code source C.4 – Grammaire TELESI de CoLin de sélection des constituants.

```
&REFER(VarCompress,GramSelect).

&GRAMMAIRE.

&ENTREE: SELECT.

// ************ DÉFINITION DE LA CATÉGORIE GRAMMATICALE DES TÊTES *************

--> SET_CGTETE.

&GRAM:SET_CGTETE(I).
TETE_PROP: 0(1) / 0:(K=PHRASE); 1:(FS=COMPCIR)&(CGTETE=0)
   => 0(1) / 1:1(CGTETE=PROP).
TETE_PRON: 0(1) / 0:((K = GN)|(K = GNPREP))&(CAT = REP); 1:(FS = ATTR)&(CGTETE=0)
   => 0(1) / 1:1(CGTETE=PRON).
TETE_NOM: 0(1) / 0:(CAT=N);
   1:(FS=ATTR)&(SOUSATTR != ATTRSUJ)&(SOUSATTR != ATTROBJ)&(TYP !$>= ADADJ)&(CGTETE=0)
   => 0(1) / 1:1(CGTETE=NOM).
TETE_ADV: 0(1) / 0:(K=GADV); 1:(FS = ATTR)&(CGTETE=0)
   => 0(1) / 1:1(CGTETE=ADV).
TETE_ADJ: 0(1) / 0:(K=GA); 1:(FS = ATTR)&(TYP $>= ADADJ)&(CGTETE=0)
   => 0(1) / 1:1(CGTETE=ADJ).
TETE_VERBE: 0(1) / 0:(K=GV); 1:(CGTETE=0)
   => 0(1) / 1:1(CGTETE=VERBE).

// ************ DÉFINITION DES COMPLÉMENTS ET MODIFIEURS *************

--> SET_COMP_MOD.

&GRAM: SET_COMP_MOD(I).
SET_MPROP: 0 / 0:(CGTETE = PROP)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=MOD).
SET_CPRON: 0(1,2) / 1:(CAT=REP)&((LEMME='celui')|(LEMME='ce')|(LEMME='un')); 2:(CGTETE = PRON)&(FSGOV = 0)
   => 0(1,2) / 2:2(FSGOV=COMP).
SET_MPRON: 0 / 0:(CGTETE = PRON)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=MOD).
SET_CADV: 0 / 0:(CGTETE = ADV)&(K = GNPREP)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=COMP).
SET_MADV: 0 / 0:(CGTETE = ADV)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=MOD).
SET_CNOM: 0 / 0:(CGTETE = NOM)&((K = GNPREP)|(KPH = PHINFPREP))&(FSGOV = 0)
   => 0 / 0:0(FSGOV=COMP).
SET_MNOM: 0 / 0:(CGTETE = NOM)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=MOD).
SET_MADJ: 0 / 0:(CGTETE = ADJ)&(SOUSA=ADVERB)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=MOD).
SET_CADJ: 0 / 0:(CGTETE = ADJ)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=COMP).
SET_CVERBE: 0 / 0:(CGTETE = VERBE)&((FS = OBJT)|(FS = OBJI))&(FSGOV = 0)
   => 0 / 0:0(FSGOV=COMP).
SET_MVERBE: 0 / 0:(CGTETE = VERBE)&(SOUSA = ADVERB)&(FSGOV = 0)
   => 0 / 0:0(FSGOV=MOD).

// ************ MODIFIEURS DE LA PROPOSITION ************
--> MPROP.
```

```
&GRAM: MPROP(I).
MPROP: 0 / 0:(SUPPRTYPE = 0)&(CGTETE=PROP)&(FSGOV=MOD)&(SEM !$>= NEGAT)&
   (((SEM !$>= INTER)&(SOUSR != REL))|((K != GADV)&(K != GN)))&(TYP !$>= OBJI)
   => 0 / 0:0(
      < SEMOBJ = LIEU :
         SUPPRTYPE = MPROPLIEU #
         < SEMOBJ = TEMPS :
            SUPPRTYPE = MPROPTEMPS #
            SUPPRTYPE = MPROPAUTRE > > ).

// ************ COMPLÉMENTS ET MODIFIEURS DU NOM ************
--> GOVNOM.

&GRAM: GOVNOM(U).
--> MNOMAPP.

// Cas des appositions nominales (détachées)
&GRAM: MNOMAPP(I).
MNOMAPP: 0(*,1) / 0:(CGTETE=NOM)&(FSGOV=MOD)&(SUPPRTYPE = 0); 1:(FRM = ',')|(FRM = '(')|(FRM = '-')
   => 0(1) / 0:0(SUPPRTYPE = MNOMAPP).
--> MNOMGENERAL.

// Cas des modifieurs du nom en général
&GRAM: MNOMGENERAL(I).
MNOMGENERAL: 0 / 0:(CGTETE=NOM)&(FSGOV=MOD)&(KPH != PHINFPREP)&
      (SEM !$>= NEGAT)&(KPH != PHREL)&(SUPPRTYPE = 0)
   => 0 / 0:0(SUPPRTYPE = MNOM).
--> CNOM.

// Cas des compléments du nom
&GRAM: CNOM(I).
CNOM: 0 / 0:(CGTETE=NOM)&(FSGOV=COMP)&(KPH!=PHREL)&(SUPPRTYPE = 0)
   => 0 / 0:0(SUPPRTYPE = CNOM).
--> MNOMSUB.

// Cas des modifieurs du nom subordonnés
&GRAM: MNOMSUB(I).
MNOMSUB: 0 / 0:(CGTETE=NOM)&(FSGOV=MOD)&(KPH = PHREL)&(SUPPRTYPE = 0)
   => 0 / 0:0(SUPPRTYPE = MNOMSUB).

// ************ MODIFIEUR DU PRONOM ************
--> MPRON.

&GRAM: MPRON(U).
--> MPRONAPP.

// Modifieur du pronom détaché
&GRAM: MPRONAPP(I).
MPRONAPP: 0(*,1) / 0:(CGTETE=PRON)&(FSGOV=MOD)&(SUPPRTYPE = 0); 1:(FRM = ',')|(FRM = '(')|(FRM = '-')
   => 0(1) / 0:0(SUPPRTYPE = MPRONAPP).

// ************ COMPLÉMENTS ET MODIFIEURS DE L'ADJECTIF ET DE L'ADVERBE ************
--> CAD.

&GRAM: CAD(I).
CADJ: 0 / 0:(CGTETE=ADJ)&(FSGOV=COMP)&(SUPPRTYPE = 0)
   => 0 / 0:0(SUPPRTYPE = CADJ).
```

```
CADV: 0 / 0:(CGTETE=ADV)&(FSGOV=COMP)&(SUPPRTYPE = 0)
   => 0 / 0:0(SUPPRTYPE = CADV).
--> MAD.

&GRAM: MAD(I).
MADJ: 0 / 0:(CGTETE=ADJ)&(FSGOV=MOD)&(SUPPRTYPE = 0)&(SEM !$>= NEGAT)
   => 0 / 0:0(SUPPRTYPE = MADJ).
MADV: 0 / 0:(CGTETE=ADV)&(FSGOV=MOD)&(SUPPRTYPE = 0)&(SEM !$>= NEGAT)
   => 0 / 0:0(SUPPRTYPE = MADV).

// ************ COMPLÉMENTS ET MODIFIEURS DU VERBE ************
--> CVERBE.

&GRAM: CVERBE(I).
COMPDVERBE: 0 / 0:(CGTETE=VERBE)&(FSGOV=COMP)&(TYP != OBJ)&(SOUSR != REFL)&(SUPPRTYPE = 0)
   => 0 / 0:0(SUPPRTYPE = CVERBE).
--> MVERBE.

&GRAM: MVERBE(I).
MVERBE: 0 / 0:(CGTETE=VERBE)&(FSGOV=MOD)&(SEM !$>= NEGAT)&(TYP !$>= OBJI)&(SUPPRTYPE = 0)
   => 0 / 0:0(SUPPRTYPE = CVERBE).

// ************ CAS DES CONSTITUANTS NON OU DIFFICILEMENT SUPPRIMABLES ************
--> ANAPHORES.

&GRAM: ANAPHORES(I).
ANAPHORES: 0 / 0:(AREP = OUI)&(VERROU !$>= V_AREP)
   => 0 / 0:0(VERROU = VERROU(0)|V_AREP).
--> CASVETAT.

&GRAM: CASVETAT(I).
CASVETAT: 0 / 0:(VERROU !$>= V_VETAT)&(FS = ATTR)&(SOUSATTR = ATTRSUJ)
   => 0 / 0:0(VERROU = VERROU(0)|V_VETAT).
--> CASVATTRSUJ.

&GRAM: CASVATTRSUJ(I).
CASVATTRSUJ: 0(1) / 0:(SOUSATTR=ATTRSUJ);
   1:(VERROU !$>= V_ATTR_SUJ)&(CGTETE=NOM)&(FSGOV = MOD)&(SUPPRTYPE != MNOMAPP)
   => 0(1) / 1:1(VERROU = VERROU(1)|V_ATTR_SUJ).
--> CASPHINT.

// cas des phrases interrogatives
&GRAM: CASPHINT(I).
CASPHINT: 0?(1) / 0:(TPH = INT); 1:(K!=0)&(VERROU !$>= V_PHINT)
   => 0(1) / 1:1(VERROU = VERROU(1)|V_PHINT).
--> CASPHNEG.

// cas des propositions négatives
&GRAM: CASPHNEG(I).
CASPHNEG_MPROP: 0(1) / 0:(ASSERT = NEG); 1:(VERROU !$>= V_PHNEG)
   => 0(1) / 1:1(VERROU = VERROU(1)|V_PHNEG).
CASPHNEG_MNOM: 0(1(2)) / 0:(ASSERT = NEG)&(K=PHRASE); 2:(CGTETE=NOM)&(FSGOV = MOD)&(VERROU !$>= V_PHNEG)
   => 0(1(2)) / 2:2(VERROU = VERROU(2)|V_PHNEG).
--> CASARTD.

// cas des articles définis/indéfinis avant les épithètes/relatives,
// je ne remonte pas la propriété de non-suppression, car on peut supprimer le GN/GNPREP en bloc
```

```
&GRAM: CASARTD(I).
CASARTD: 0(1,2) /
   0:(CAT = N);
   1:((CAT = DETERM)|(CAT = PREP))&(SOUSD !$>= ARTI);
   2:((SUPPRTYPE = CNOM)|(SUPPRTYPE = MNOM)|(SUPPRTYPE = MNOMSUB))&
         (SUPPRTYPE != MNOMAPP)&(VERROU !$>= V_ARTD)
   => 0(1,2) /
   2:2(VERROU = VERROU(2)|V_ARTD).

// ************ EXCEPTION DES PARENTHÈSES ************
--> PARENTHESES.

&GRAM: PARENTHESES(I).
PARENTHESES: 0(*,1,2,*) / 0:(VERROU != 0); 1:(CATPONCT=PARENTHESE); 2:(CATPONCT=PARENTHESE)
      => 0(1,2) / 0:0(VERROU = 0).

// ************ PROPAGATION DES VERROUX ************
// une fois tous les constituants verrouillés (non supprimables) définis, je remonte la propriété de
// non-suppression de fils en père pour lesquels c'est nécessaire et je supprime les valeurs de
// SUPPRTYPE où VERROU est non nul
--> REMONTE_VERROU.

&GRAM: REMONTE_VERROU(I).
REMONTEVERROU: 0(1) / 0:(VERROU !$>= V_AREP)&(LEMME != ''); 1:(VERROU $>= V_AREP)
   => 0(1) / 0:0(VERROU = VERROU(0)|V_AREP).
--> PRIORITE_VERROU.

&GRAM: PRIORITE_VERROU(I).
PRIORITE_VERROU: 0 / 0:(VERROU $>= V_AREP)&(VERROU $>= V_ARTD)
   => 0 / 0:0(VERROU = V_ARTD).
--> APPL_VERROU.

// applique les verrous obligatoires
&GRAM: APPL_VERROU(I).
APPL_VERROU: 0 / 0:((VERROU $>= V_CPRON)|(VERROU $>= V_VETAT))&(SUPPRTYPE != 0)
   => 0 / 0:0(SUPPRTYPE = 0).
--> %STOP.

&FIN.
```

Code source C.5 – Extrait de la grammaire TELESI de CoLin de préparation à la linéarisation finale.

```
&REFER(VarCompress,GramArrange).

&GRAMMAIRE.

&ENTREE: TRAITEMENT.

// [...]

--> SETDEBFINCONST.

// définit où commence et finit un constituant, afin de définir la place des balises de suppression
&GRAM: SETDEBFINCONST(I).
SETDEBCONSTFEUILLE: 0(1(*)) / 1:(PLACEMOT!=0) / ((DEBCONST(0)=0)|(DEBCONST(0)>PLACEMOT(1)))
```

```
   => 0(1) / 0:0(DEBCONST=PLACEMOT(1); DECDEBCONST=-1).
SETFINCONSTFEUILLE: 0(1(*)) / 1:(PLACEMOT!=0) / ((FINCONST(0)=0)|(FINCONST(0)<PLACEMOT(1)))
   => 0(1) / 0:0(FINCONST=PLACEMOT(1); DECFINCONST=1).
SETDEBCONST: 0(1) / 1:(DEBCONST!=0) / ((DEBCONST(0)=0)|(DEBCONST(0)>DEBCONST(1)))
   => 0(1) / 0:0(DEBCONST=DEBCONST(1); DECDEBCONST=DECDEBCONST-1).
SETFINCONST: 0(1) / 1:(FINCONST!=0) / ((FINCONST(0)=0)|(FINCONST(0)<FINCONST(1)))
   => 0(1) / 0:0(FINCONST=FINCONST(1); DECFINCONST=DECFINCONST+1).
--> AREPAAF.

// encadre les constituants référés
&GRAM: AREPAAF(I).
AREPAAF: 0(*,1) / 0:(AREP != 0); 1:(BALISE = 0)
   => x(A,1,*0<1,>*,B) /
      x:0;
      A:(FRM = '<div name="arep" style="display:inline;">'; BALISE = AUTRE; FORMEBALISE = OUVRANTE;
         PLACEMOT=DEBCONST(0); DECDEBCONST=DECDEBCONST(0));
      B:(FRM = '</div>'; BALISE = AUTRE; FORMEBALISE = FERMANTE;
         PLACEMOT=FINCONST(0); DECFINCONST=DECFINCONST(0)).
--> AJBALISESSUPPR.

// encadre les constituants à supprimer
&GRAM: AJBALISESSUPPR(I).
AJSUPPR: 0(*,1) / 0:(SUPPRTYPE != 0); 1:(BALISE != MOT)
   => x(A,1,*0<1,>*,B) /
      x:0;
      A:(FRM = '<span name="mot" type="'||%CHAINE(SUPPRTYPE(0))||'" verrou="'||%CHAINE(VERROU(0))||'">';
         BALISE = MOT; SUPPRTYPE=SUPPRTYPE(0); FORMEBALISE = OUVRANTE; PLACEMOT=DEBCONST(0);
         DECDEBCONST=DECDEBCONST(0));
      B:(FRM = '</span>'; BALISE = MOT; SUPPRTYPE=SUPPRTYPE(0); FORMEBALISE = FERMANTE;
         PLACEMOT=FINCONST(0); DECFINCONST=DECFINCONST(0)).
--> AJBALISESNONSUPPR.

// encadre les constituants à ne pas supprimer
&GRAM: AJBALISESNONSUPPR(U).
AJNONSUPPR: *(0(1))
   => x(A,*0<,1>*,1,*0<1,>*,B) /
      x:0;
      A:(FRM = '<span name="mot" type="NONSUPPR">'; BALISE = MOT; FORMEBALISE = OUVRANTE);
      B:(FRM = '</span>'; BALISE = MOT; FORMEBALISE = FERMANTE).
--> PLACEBALISE.

// place les balises sous chaque feuille
&GRAM: PLACEBALISE(I,PLACEBALISE,PLACEBALISE).
RPLACE: *(0(*,1,2?(3(*)),4,*)) / 1:(BALISE = MOT); 3:(BALISE = 0); 4:(BALISE = MOT)
   => 0(1,2(3(X,Y)),4) /
      X:(BALISE = MOT; SUPPRTYPE = SUPPRTYPE(1); FORMEBALISE = FORMEBALISE(1); FRM = FRM(1));
      Y:(BALISE = MOT; SUPPRTYPE = SUPPRTYPE(4); FORMEBALISE = FORMEBALISE(4); FRM = FRM(4)).
RPLACE2:*(0(*,1,3(*),4,*)) / 1:(BALISE = MOT); 3:(BALISE = 0); 4:(BALISE = MOT)
   => 0(1,3(X,Y),4) /
      X:(BALISE = MOT; SUPPRTYPE = SUPPRTYPE(1); FORMEBALISE = FORMEBALISE(1); FRM = FRM(1));
      Y:(BALISE = MOT; SUPPRTYPE = SUPPRTYPE(4); FORMEBALISE = FORMEBALISE(4); FRM = FRM(4)).
--> DEFCONST.

// définit les constituants supprimables
&GRAM: DEFCONST(I).
DEFCONST: 0(*,1,2(3),4,*) / 1:(BALISE = MOT); 4:(BALISE = MOT)
   => 0(1,2(3),4) / 1:1(
```

```
      BALISE = CONSTI;
      FRM=%TCHAINE(FRM,'"mot"','"constituant"');
      FRM=%TCHAINE(FRM,'span','div'));
   4:4(BALISE = CONSTI; FRM=%TCHAINE(FRM,'span','div')).
--> ARRANGE.

&GRAM: ARRANGE(U,PARCOURS).
--> SUPPRBALISESPONCT.

// applatit l'arbre et réordonne les feuilles
&GRAM: PARCOURS(I,,PARCOURS).
RMONTE: *(0(1(*,2,3))) / 2:(BALISE != MOT) => 0(2,1(3)).
RDEMONTE: *(0(1(*,2,*))) / 2:(BALISE != MOT) => x(*0<,1>*,2,*0<1,>*) / x:0.
RORDRE1: 0,1 / 0:(BALISE != MOT)&(PLACEMOT != 0); 1:(BALISE != MOT)&(PLACEMOT != 0) /
   PLACEMOT(0) > PLACEMOT(1)
   => x(*1*),*@<0,1>*,y(*0*) / x:1 ; y:0.
RORDRE2: 0,1 / 0:(BALISE != MOT)&(PLACEMOT != 0); 1:(BALISE != MOT)&(PLACEMOT != 0) /
   (PLACEMOT(0) = PLACEMOT(1))&(DECDEBCONST(0)+DECFINCONST(0)>DECDEBCONST(1)+DECFINCONST(1))
   => x(*1*),*@<0,1>*,y(*0*) / x:1 ; y:0.
--> %STOP.

// ajoute les balises pour les ponctuations (pour éviter de les compter comme mots dans les stats)
&GRAM: SUPPRBALISESPONCT(U).
SUPPRBALISESPONCT: 0(*,1,*,2,*) / 0:(CAT = PONCT); 1:(BALISE = MOT)&(FORMEBALISE != 0);
   2:(BALISE = MOT)&(FORMEBALISE != 0)
   => 0(1,2) / 1:1(FRM='<span name="mot" type="ponct">'); 2:2(FRM='</span>').
--> RMONTEBALISES.

// replace les balises de part et d'autre des feuilles, plutôt que dessous
&GRAM: RMONTEBALISES(I).
RMONTEBALISES: 0(*,1,*,2,*) / 1:(BALISE = MOT); 2:(BALISE = MOT)
   => 1,0,2.
--> SET_PARA.

// définit la variable pour les changements de paragraphes
&GRAM: SET_PARA(I).
SET_PARA: 0(1,*,2) / 1:(BALISE = MOT)&(RETOUR = 0); 2:(PARAGRAPH = 1)
   => 0(1,2) / 1:1(FRM=%TCHAINE(FRM,'<span','<span retour="2"'); RETOUR = 2); 2:2(CHGMTLG = 0).
--> SET_CHGMTLG.

// définit la variable pour les retours à la ligne
&GRAM: SET_CHGMTLG(I).
SET_CHGMTLG: 0(1,*,2) / 1:(BALISE = MOT)&(RETOUR = 0); 2:(CHGMTLG = 1)
   => 0(1,2) / 1:1(FRM=%TCHAINE(FRM,'<span','<span retour="1"');RETOUR = 1).
--> PROPA_RETOUR.

&GRAM: PROPA_RETOUR(I).
PROPA_RETOUR: 0(1,*,2) / 1:(FORMEBALISE = OUVRANTE)&(RETOUR = 0); 2:(RETOUR != 0)
   => 0(1,2) / 1:1(RETOUR = RETOUR(2)); 2:2(RETOUR = 0).
--> SET_GENRE_ARTD.

// spécifie le genre des articles afin de restituer les lettres élisées dans certaines compressions
&GRAM: SET_GENRE_ARTD(I).
SET_GENRE_ARTD_MAS: 0(1,*,2) / 1:(FORMEBALISE = OUVRANTE); 2:(SOUSD=ARTD)&(GNR $>= MAS)
   => 0(1,2) / 1:1(FRM=%TCHAINE(FRM,'<span','<span genre="MAS"')); 2:2(GNR=0).
SET_GENRE_ARTD_FEM: 0(1,*,2) / 1:(FORMEBALISE = OUVRANTE); 2:(SOUSD=ARTD)&(GNR $>= FEM)
   => 0(1,2) / 1:1(FRM=%TCHAINE(FRM,'<span','<span genre="FEM"')); 2:2(GNR=0).
```

```
--> ADD_BR.

// ajoute les balises de changement de paragraphe et de retour à la ligne
&GRAM: ADD_BR(I).
ADD_BR1: 0(1) / 1:(RETOUR = 1)
   => x(*0<,1>*,A,1,*0<1,>*) /
      x:0;
      A:(FRM='<br/>');
      1:1(RETOUR = 0).
ADD_BR2: 0(1) / 1:(RETOUR = 2)
   => x(*0<,1>*,A,1,*0<1,>*) /
      x:0;
      A:(FRM='<br/><br/>');
      1:1(RETOUR = 0).
--> BALISESVIDES.

&GRAM: BALISESVIDES(I).
// supprime les balises vides, <> </>
BALISESVIDES: 3,*,1,*,2,*,4 / 2:(FORMEBALISE = FERMANTE);
   1:(FORMEBALISE = OUVRANTE) / SUPPRTYPE(1) = SUPPRTYPE(2)
   => 3,4.
// joint les ponctuations liées à un mot
JOINTPONCTMOTS: 3,*,1,*,2,*,4 / 1:(BALISE != 0)&(FORMEBALISE = 0); 2:(BALISE != 0)&(FORMEBALISE = 0)
   => 3,4.
--> %STOP.

&FIN.
```

Bibliographie

[Alonso & Fort, 2003] Laura Alemany Alonso et Maria Fuentes Fort. « Integrating Cohesion and Coherence for Automatic Summarization ». Dans les actes de *EACL03*, Budapest, Hungary, April 2003.

[Alonso *et al.*, 2003a] Laura Alemany Alonso, Bernardino Casas, Irene Castellón, Salvador Climent, et Lluís Padró. « Combining heterogeneous knowledge sources in e-mail summarization ». Dans les actes de *Recent Advances in Natural Language Processing (RANLP 2003)*, pp 10–12, Borovets in Bulgaria, September 2003.

[Alonso *et al.*, 2003b] Laura Alemany Alonso, Irene Castellón, Salvador Climent, Maria Fuentes, Lluís Padró, et Horacio Rodríguez. « Approaches to Text Summarization : Questions and Answers ». *Revista Iberoamericana de Inteligencia Artificial*, pp 34–52, 2003.

[Ando *et al.*, 2000] Rie Kubota Ando, Branimir K. Boguraev, Roy J. Byrd, et Mary S. Neff. « Multi-document summarization by visualizing topical content ». Dans les actes de *ANLP/NAACL 2000 Workshop on Automatic Summarization*, 2000.

[Aone *et al.*, 1998] Chinatsu Aone, Mary Ellen Okurowski, et James Gorlinsky. « Trainable, Scalable Summarization Using Robust NLP and Machine Learning ». Dans les actes de *the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics*, pp 62–66, 1998.

[Azzam *et al.*, 1999] Saliha Azzam, Kevin Humphreys, et Robert Gaizauskas. « Using coreference chains for text summarization ».

Dans les actes de *the ACL Workshop on Coreference and its Applications, Maryland, 1999.*, 1999.

[Baldwin & Morton, 1998] Breck Baldwin et Thomas S. Morton. « Dynamic coreference-based summarization ». Dans les actes de *EMNLP-3 Conference*, 1998.

[Barzilay & Elhadad, 1997] Regina Barzilay et Michael Elhadad. « Using lexical chains for text summarization ». Dans les actes de *the Intelligent Scalable Text Summarization Workshop (ISTS'97)*, Madrid, Spain, 1997. ACL.

[Bes1990] *Bescherelle 3 - La grammaire pour tous.* Bescherelle. Hatier Paris, 1990.

[Black, 1996] Cheryl A. Black. « A step-by-step introduction to Government and Binding theory of syntax ». Dans les actes de *Notes on Linguistics*, volume 73, pp 5–12, 1996. `http://www.sil.org/mexico/ling/E002-IntroGB.htm`.

[Boguraev & Neff, 2000] Branimir K. Boguraev et Mary S. Neff. « Lexical Cohesion, Discourse Segmentation and Document Summarization ». Dans les actes de *RIAO-2000*, 2000.

[Bouiller *et al.*, 2005] Pierre Bouiller, Lionel Clément, Benoît Sagot, et Éric Villemonte de la Clergerie. « Simple comme EASy :-) ». Dans les actes de *Proceedings of the EASy Workshop in TALN 05*, pp 57–60, 2005.

[Carson *et al.*, 2002] Chad Carson, Serge Belongie, Hayit Greenspan, et Jitendra Malik. « Blobworld : Image segmentation using expectation-maximization and its application to image querying ». *IEEE Trans. Pattern Anal. Mach. Intell.*, pp 1026–1038, 2002.

[Chang *et al.*, 2002] Peng Chang, Mei Han, et Yihong Gong. « Extract highlights from baseball game video with hidden markov models ». Dans les actes de *the International Conference on Image Processing (ICIP '02)*, 2002.

[Chauché, 1984] Jacques Chauché. « Un outil multidimensionnel de l'analyse du discours ». Dans les actes de *Coling'84*, pp 11–15, 1984.

[Chauché, 2001] Jacques Chauché. « *SYGMART : Manuel de référence,*

Version 4.0 », 2001. `http://www.lirmm.fr/~chauche/REFERENCESYG/`.

[Chauché *et al.*, 2003] Jacques CHAUCHÉ, Violaine PRINCE, Simon JAILLET, et Maguelone TEISSEIRE. « Classification automatique de textes à partir de leur analyse syntaxico-sémantique ». Dans les actes de *TALN'2003*, volume 1, pp 45–55, Batz-sur-mer, 2003.

[Chaves, 2001] Rui Pedro CHAVES. « WordNet and Automated Text Summarization ». Dans les actes de *the 6th Natural Language Processing Pacific Rim Symposium, NLPRS*, Tokyo, Japan, 2001.

[Chomsky, 1970] Noam CHOMSKY. « Remarks on nominalization ». Dans les actes de *R. Jacobs and P. Rosenbaum (eds.) Reading in English Transformational Grammar*, pp 184–221, Waltham : Ginn, 1970.

[Chomsky, 1981] Noam CHOMSKY. *Lectures on Government and Binding.* Foris Publications, Dordrecht, Netherlands, 1981.

[Chomsky, 1982] Noam CHOMSKY. *Some Concepts and Consequences of the Theory of Government and binding.* Linguistic Inquiry monograph n ° 6, MIT Press, Cambridge, Mass., 1982.

[Chomsky, 1986] Noam CHOMSKY. *Barriers.* MIT Press, Cambridge, 1986.

[Coldefy *et al.*, 2004] François COLDEFY, Patrick BOUTHEMY, Michaël BETSER, et Guillaume GRAVIER. « Tennis video abstraction from audio and visual cues ». Dans les actes de *IEEE International Workshop on Multimedia Signal Processing, MMSP'2004*, Siene, Italie, September 2004.

[Collins, 1997] Michael COLLINS. « Three generative lexicalized models for statistical parsing ». Dans les actes de *the 35th Annual Meeting of the Association for Computational Linguistics (ACL-97)*, pp 16–23, Madrid, Spain, 1997.

[Daumé III *et al.*, 2002] Hal DAUMÉ III, Abdessamad ECHIHABI, Daniel MARCU, Dragos Stefan MUNTEANU, et Radu SORICU. « GLEANS : A Generator of Logical Extracts and Abstracts for Nice Summaries ». Dans les actes de *the Document Understanding Conference (DUC-2002)*, Philadelphia, PA, July 2002.

[Deerwester *et al.*, 1990] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, et Richard A. Harshman. « Indexing by Latent Semantic Analysis ». *Journal of the American Society of Information Science*, pp 391–407, 1990.

[Donaway *et al.*, 2000] Robert L. Donaway, Kevin W. Drummey, et Laura A. Mather. « A comparison of rankings produced by summarization evaluation measures ». Dans les actes de *NAACL-ANLP 2000 Workshop on Automatic summarization*, pp 69–78, 2000.

[Dunning, 1994] Ted Dunning. « Accurate Methods for the Statistics of Surprise and Coincidence ». *Computational Linguistics*, pp 61–74, 1994.

[Ekin *et al.*, 2003] Ahmet Ekin, A. Murat Tekalp, et Rajiv Mehrotra. « Automatic soccer video analysis and summarization ». *IEEE Trans. on Image Processing*, pp 796–807, July 2003.

[Elhadad & Robin, 1996] Michael Elhadad et Jacques Robin. « An overview of SURGE : a re-usable comprehensive syntactic realization component ». Dans les actes de *the 8th International Workshop on Natural Language generation (demonstration session) (INLG'96)*, Brighton, UK, 1996.

[Erkan & Radev, 2004] Güneş Erkan et Dragomir R. Radev. « LexRank : Graph-based Centrality as Salience in Text Summarization ». Dans les actes de *Journal of Artificial Intelligence Research (JAIR)*, 2004.

[Farzindar & Lapalme, 2005] Atefeh Farzindar et Guy Lapalme. « Production automatique du résumé de textes juridiques : évaluation de qualité et d'acceptabilité ». *TALN 2005*, pp 183–192, June 2005.

[Fei-Fei *et al.*, 2003] Li Fei-Fei, Rob Fergus, et Pietro Perona. « A bayesian approach to unsupervised one-shot learning of object categories ». Dans les actes de *Ninth IEEE International Conference on Computer Vision (ICCV'03)*, volume 2, pp 1134–1141, 2003.

[Fiszman *et al.*, 2004] Marcelo Fiszman, Thomas C. Rindflesch, et Halil Kilicoglu. « Abstraction Summarization for Managing the

Biomedical Research Literature ». Dans les actes de *the HLT-NAACL Workshop on Computational Lexical Semantics*, pp 76–83, Boston, MA, North American Association for Computational Linguistics, 2004.

[Fuentes & Rodríguez, 2002] Maria FUENTES et Horacio RODRÍGUEZ. « Using cohesive properties of text for Automatic Summarization ». Dans les actes de *the Primeras Jornadas de Tratamiento y Recuperación de Información (JOTRI2002)*, Valencia, Spain, 2002.

[Goldstein *et al.*, 2000] Jade GOLDSTEIN, Vibhu MITTAL, Jaime CARBONELL, et Mark KANTROWITZ. « Multi-document summarization by sentence extraction ». Dans les actes de *Hahn et al.[15]*, pp 40–48, 2000.

[Grefenstette, 1998] Gregory GREFENSTETTE. « Producing intelligent telegraphic text reduction to provide audio scanning service for the blind ». Dans les actes de *AAAI symposium on Intelligent Text Summarisation*, pp 111–117, Menlo Park, California, 1998.

[Grevisse, 1993–1997] Maurice GREVISSE. *Le Bon Usage – Grammaire française*. édition refondue par André Goosse, DeBoeck-Duculot, Paris – Louvain-la-Neuve, 13e édition, ISBN 2-8011-1045-0, 1993–1997.

[Gross, 1975] Maurice GROSS. *Méthodes en syntaxe*. Hermann, Paris, 1975.

[Harabagiu & Lăcătuşu, 2002] Sanda M. HARABAGIU et Finley LĂCĂTUŞU. « Generating Single and Multi-Document Summaries with GISTEXTER ». Dans les actes de *DUC02-WS*, Philadelphia, PA, July 2002.

[Harabagiu *et al.*, 2003] Sanda M. HARABAGIU, V. Finley LĂCĂTUŞU, et Steven J. MAIORANO. « Multi-Document Summaries Based on Semantic Redundancy ». Dans les actes de *the 14th Florida AI Conference, (FLAIRS-2003)*, pp 387–391, St. Augustine FL, 2003.

[Hirao *et al.*, 2002] Tsutomu HIRAO, Hideki ISOZAKI, Eisaku MAEDA, et Yuji MATSUMOTO. « Extracting Important Sentences with Support Vector Machines ». Dans les actes de *the 19th*

International Conference on Computational Linguistics (COLING 2002), pp 342–348, Taipei, Taiwan, August 2002.

[Hori & Furui, 2001] Chiori HORI et Sadaoki FURUI. « Advances in automatic speech summarization ». *Eurospeech 2001*, pp 1771–1774, 2001.

[Hovy *et al.*, 2005] Eduard H. HOVY, Chin-Yew LIN, et Liang ZHOU. « A BE-based Multi-document Summarizer with Sentence Compression ». Dans les actes de *the Multilingual Summarization Evaluation Workshop at the ACL 2005 conference*, 2005.

[Inoue *et al.*, 2004] Akira INOUE, Takayoshi MIKAMI, et Yoichi YAMASHITA. « Improvement of speech summarization using prosodic information ». Dans les actes de *SP-2004*, pp 599–602, 2004.

[Ishikawa *et al.*, 2002] Kai ISHIKAWA, Shin ichi ANDO, Shin ichi DOI, et Akitoshi OKUMURA. « Trainable Automatic Text Summarization Using Segmentation of Sentence ». Dans les actes de *the Third NTCIR Workshop on research in information Retrieval, Automatic Text Summarization and Question Answering*, 2002.

[Jing, 2000] Hongyan JING. « Sentence Reduction for Automatic Text Summarization ». Dans les actes de *the 6th Conference on Applied Natural Language Processing*, pp 310–315, 2000.

[Jing *et al.*, 1998] Hongyan JING, Regina BARZILAY, Kathleen MCKEOWN, et Michael ELHADAD. « Summarization evaluation methods experiments and analysis ». 1998.

[Joshi, 1987] Aravind K. JOSHI. « *An Introduction to Tree Adjoining Grammars* », pp 87–115. Manaster-Ramer, A. (editor), Mathematics of Language. John Benjamins Publishing Co., Amsterdam/Philadelphia, 1987.

[Kaplan & Bresnan, 1982] Ronald. M.. KAPLAN et Joan BRESNAN. « *Lexical-Functional Grammar : A formal system for grammatical representation* », pp 173–281. Joan Bresnan (editor), The Mental Representation of Grammatical Relations. The MIT Press, Cambridge, MA, 1982. Reprinted

in Dalrymple et al. (editors), Formal Issues in Lexical-Functional Grammar. CSLI.

[Knight & Marcu, 2000] Kevin KNIGHT et Daniel MARCU. « Statistics-Based Summarization - Step One : Sentence Compression ». Dans les actes de *the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pp 703–710, Sapporo, Japan, 2000.

[Knight & Marcu, 2002] Kevin KNIGHT et Daniel MARCU. « Summarization beyond sentence extraction : a probabilistic approach to sentence compression ». *Artificial Intelligence archive*, pp 91–107, July 2002.

[Koenemann & Belkin, 1996] Jurgen KOENEMANN et Nicholas J. BELKIN. « A Case for Interaction : A Study of Interactive Information Retrieval Behavior and Effectiveness ». Dans les actes de *the SIGCHI conference on Human factors in computing systems : common ground, CHI '96*, pp 205–212, Vancouver, British Columbia, Canada, 1996.

[Landragin, 2004] Frédéric LANDRAGIN. « Saillance physique et saillance cognitive ». *Corela*, December 2004. `http://edel.univ-poitiers.fr/corela/document.php?id=142`.

[Leuski *et al.*, 2003] Anton LEUSKI, Chin Yew LIN, , et Eduard H. HOVY. « iNeATS : Interactive Multi-Document Summarization ». Dans les actes de *the 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pp 125–128, Sapporo, Japan, July 2003.

[Lin & Hovy, 2000] Chin-Yew LIN et Eduard H. HOVY. « The Automated Acquisition of Topic Signatures for Text Summarization ». Dans les actes de *Proceedings of COLING 2000.*, Strasbourg, France, August 2000.

[Lin & Hovy, 2002] Chin-Yew LIN et Eduard H. HOVY. « Automated Multi-Document Summarization in NeATS ». Dans les actes de *the DARPA Human Language Technology Conference*, pp 50–53, 2002.

[Lin *et al.*, 2003] Chin Yew LIN, , et Eduard H. HOVY. « Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics

». Dans les actes de *the Human Technology Conference 2003 (HLT-NAACL-2003)*, pp 150–156, 2003.

[Lucien, 1959] Tesnière Lucien. *Éléments de syntaxe structurale*. Klincksieck, Paris, 1959.

[Luhn, 1958] Hans Peter Luhn. « The automatic creation of literature abstracts ». *IBM Journal of research and development*, pp 159–165, 1958.

[Mani & Wilson, 2000] Inderjeet Mani et George Wilson. « Robust temporal processing of news ». Dans les actes de *the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*, pp 69–76, New Brunswick, New Jersey, 2000. Association for Computational Linguistics.

[Mani, 2004] Inderjeet Mani. « Narrative Summarization ». *TAL, Résmé automatique de textes*, pp 15–38, 2004.

[Mann & Thompson, 1987] William C. Mann et Sandra A. Thompson. « Rhetorical Structure Theory : a Framework for the Analysis of Texts ». Dans les actes de *Technical Report ISI/RS-87-185, Information Sciences Institute*, Marina del Rey, California, 1987.

[Marcu, 1998] Daniel Marcu. « Improving summarization through rhetorical parsing tuning ». Dans les actes de *the COLING-ACL Workshop on Very Large Corpora*, Montreal, Canada, 1998.

[Mauffrey & Cohen, 1995] Annick Mauffrey et Isdey Cohen. *La grammaire française*. Hachette Education, 3ème édition, 1995.

[McCord, 1990] Michael McCord. « English Slot Grammar ». Dans les actes de *IBM*, 1990.

[McKeown & Radev, 1995] Kathleen R. McKeown et Dragomir R. Radev. « Generating Summaries of Multiple News Articles ». Dans les actes de *Proceedings, 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp 74–82, Seattle, Washington, 1995.

[McKeown *et al.*, 1999] Kathleen McKeown, Judith Klavans, Vasileios Hatzivassiloglou, Regina Barzilay, et Eleazar Eskin. « Towards Multidocument Summarization by Reformulation :

Progress and Prospects ». Dans les actes de *Proceedings of AAAI*, pp 453–460, 1999.

[McKeown *et al.*, 2001] Kathleen R. McKeown, Regina Barzilay, David Evans, Vasileios Hatzivassiloglou, Barry Schiffman, et Simone Teufel. « Columbia Multi-Document Summarization : Approach and Evaluation ». Dans les actes de *the Workshop on Text Summarization, ACM SIGIR Conference*. DARPA/NIST, Document Understanding Conference, 2001.

[Mela, 2007] Auguta Mela. « Le complément en linguistique ». Rapport interne, LIRMM, 2007.

[Mel'čuk *et al.*, 1995] Igor A. Mel'čuk, André Clas, et Alain Polguère. « *Introduction à la Lexicologie Explicative et Combinatoire* », pp 125–138. Duculot, 1995.

[Miller *et al.*, 1990] George Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, et Katherine J. Miller. « Introduction to wordnet : an on-line lexical database ». *International Journal of Lexicography*, pp 235–244, 1990.

[Minel, 2004] Jean-Luc Minel. « Le résumé automatique de textes : solutions et perspectives ». *TAL, Résumé automatique de textes*, pp 7–13, 2004.

[Minel, 2007] Jean-Luc Minel. « Qu'est-ce que le résumé automatique ? ». Dans les actes de *Propos recueillis par Richard Walter pour le laboratoire CRIS - Université Paris X*, June 2007. `http://www.technolangue.net/article.php3?id_article=329`.

[Minel *et al.*, 2001] Jean-Luc Minel, Jean-Pierre Desclés, Emmanuel Cartier, Gustavo Crispino, Slim Ben Hazez, et Agata Jackiewicz. « Résumé automatique par filtrage sémantique d'informations dans des textes. Présentation de la plateforme FilText ». *Revue Technique et Science Informatique*, 2001.

[Oka & Ueda, 2001] Mamiko Oka et Yoshihiro Ueda. « Phrase-representation Summarization Method and Its Evaluation ». Dans les actes de *the Second NTCIR Workshop on Research in Chinese & Japanese Text Retrieval and Text Summarization*, Tokyo, Japan, March 2001.

[Papineni *et al.*, 2002] Kishore PAPINENI, Salim ROUKOS, Todd WARD, et Wei-Jing ZHU. « Bleu : a method for automatic evaluation of machine translation ». Dans les actes de *the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp 311–318, July 2002.

[Pollard & Sag, 1994] Carl POLLARD et Ivan A. SAG. *Head-Driven Phrase Structure Grammar*. University of Chicago Press and CSLI Publications, Chicago, Illinois, 1994.

[Pugeault, 1995] Florence PUGEAULT. *Extraction dans les textes de connaissances structurées : une méthode fondée sur la sémantique lexicale*. Thèse de l' Université Paul Sabatier, Toulouse 3, n° 2153, 1995.

[Quinlan, 1993] J. Ross QUINLAN. « C4.5 : Programs for Machine Learning ». Dans les actes de *Morgan Kaufmann*, San Mateo, CA, 1993.

[Radev & McKeown, 1998] Dragomir R. RADEV et Kathleen MCKEOWN. « Generating Natural Language Summaries from Multiple On-Line Sources ». *Computational Linguistics*, pp 469–500, 1998.

[Radev *et al.*, 2004] Dragomir R. RADEV, Hongyan JING, Malgorzata STYŚ, et Daniel TAM. « Centroid-based summarization of multiple documents ». Dans les actes de *DUC 2003*, pp 919–938, 2004.

[Rastier, 2002] François RASTIER. « La macrosémantique, Petite introduction didactique à la sémantique textuelle ». `http://www.revue-texto.net/Inedits/Rastier/Rastier_Macrosemantique1.html`, 2002.

[Saggion & Lapalme, 2002] Horacio SAGGION et Guy LAPALME. « Generating indicative-informative summaries with sumUM ». *Computational Linguistics*, pp 497–526, 2002.

[Sagot & Danlos, 2006] Benoît SAGOT et Laurence DANLOS. « Améliorer un lexique syntaxique à l'aide des tables du lexique-grammaire ». Dans les actes de *Colloque DLTAF 2006 (Description Linguistique pour le Traitement Automatique du Français) du congres de l'ACFAS*, Montréal, Canada, 2006.

[Sagot *et al.*, 2005] Benoît SAGOT, Lionel CLÉMENT, Éric Villemonte de la CLERGERIE, et Pierre BOUILLER. « Vers un méta-lexique

pour le français : architecture, acquisition, utilisation ». Dans les actes de *Journée ATALA sur l'interface lexique-grammaire*, 2005.

[Salton & Yang, 1973] Gerard SALTON et Chung S. YANG. « On the specification of term values in automatic indexing ». Dans les actes de *Journal of Documentation 29*, pp 351–372, April 1973.

[Schneider, 1998] Gerold SCHNEIDER. « An Introduction to Government & Binding ». Englisches Seminar der Universität Zürich, `http://www.ifi.unizh.ch/CL/gschneid/dreitaegig.pdf`, 1998.

[Schwab, 2005] Didier SCHWAB. « *Approche hybride - lexicale et thématique - pour la modélisation, la détection et l'exploitation des fonctions lexicales en vue de l'analyse sémantique de texte* ». PhD thesis, Université Montpellier II, Montpellier, December 2005.

[Schwab *et al.*, 2005] Didier SCHWAB, Mathieu LAFOURCADE, et Violaine PRINCE. « Extraction semi-supervisée de couples d'antonymes grâce à leur morphologie ». Dans les actes de *TALN'2005 : Traitement Automatique des Langues Naturelles*, pp 73–82, Dourdan, France, June 2005.

[Somers *et al.*, 1997] Harold SOMERS, Bill BLACK, Joakim NIVRE, Torbjörn LAGER, Annarosa MULTARI, Luca GILARDONI, Jeremy ELLMAN, et Alex ROGERS. « Multilingual generation and summarization of job adverts : the TREE project ». Dans les actes de *the 5th. Conference on Applied Natural Language Processing*, pp 269–276, Washington, D.C., 1997.

[Tesnière, 1934] Lucien TESNIÈRE. « Comment construire une syntaxe ». Dans les actes de *Bulletin de la Faculté des Lettres de Strasbourg, 7, 12ème année*, pp 219–229, 1934.

[TLF2007] « Le Trésor de la Langue Française informatisé (TLFi) ». `http://atilf.atilf.fr`, August 2007.

[Tomassone, 2001] Roberte TOMASSONE. « A propos des "compléments circonstanciels" ». Dans les actes de *Les revues pédagogiques de la Mission Laïque Française*, pp 43–59, Novembre 2001.

[Vergne, 2001] Jacques VERGNE. « Analyse syntaxique automatique de langues : du combinatoire au calculatoire (communication

invitée) ». Dans les actes de *Proceedings of TALN 2001*, pp 15–29, 2001.

[Wagner & Pinchon, 1962] Robert Léon Wagner et Jacqueline Pinchon. *Grammaire du Français classique et moderne*. Hachette Université, Paris, 1962.

[Wan *et al.*, 2003] Stephen Wan, Robert Dale, Mark Dras, et Cécile Paris. « Straight to the Point : Discovering Themes for Summary Generation ». Dans les actes de *the Australian Workshop on Natural Language Processing*, Melbourne, Australia, 2003.

[White *et al.*, 2001] Michael White, Tanya Korelsky, Claire Cardie, Vincent Ng, David Pierce, et Kiri Wagstaff. « Multidocument Summarization via Information Extraction ». Dans les actes de *HLT 2001, Human Language Technology Conference*, San Diego, CA, 2001.

[Yousfi-Monod & Prince, 2005a] Mehdi Yousfi-Monod et Violaine Prince. « Automatic summarization based on sentence morpho-syntactic structure : narrative sentences compression ». Dans les actes de *the 2nd International Workshop on Natural Language Understanding and Cognitive Science (NLUCS 2005)*, pp 161–167, Miami/USA, May 2005.

[Yousfi-Monod & Prince, 2005b] Mehdi Yousfi-Monod et Violaine Prince. « Utilisation de la structure morpho-syntaxique des phrases dans le résumé automatique ». Dans les actes de *TALN 2005*, pp 193–202, Dourdan, June 2005.

[Yousfi-Monod & Prince, 2006] Mehdi Yousfi-Monod et Violaine Prince. « Compression de phrases par élagage de l'arbre morpho-syntaxique ». *Technique et Science Informatiques*, pp 437–468, 2006.

Résumé : Le travail s'inscrit dans le domaine du traitement automatique du langage naturel et traite plus spécifiquement d'une application de ce dernier au résumé automatique de textes. L'originalité de la thèse consiste à s'attaquer à une variété fort peu explorée, la compression de textes, par une technique non supervisée. Ce travail propose un système incrémental et interactif d'élagage de l'arbre syntagmatique des phrases, tout en préservant la cohérence syntaxique et la conservation du contenu informationnel important. Sur le plan théorique, le travail s'appuie sur la théorie du gouvernement de Noam Chomsky et plus particulièrement sur la représentation formelle de la théorie X-barre pour aboutir à un fondement théorique important pour un modèle computationnel compatible avec la compression syntaxique de phrases. Le travail a donné lieu a un logiciel opérationnel, nommé CoLin, qui propose deux modalités : une compression automatique, et une aide au résumé sous forme semi-automatique, dirigée par l'interaction avec l'utilisateur. Le logiciel a été évalué grâce à un protocole complexe par 25 utilisateurs bénévoles. Les résultats de l'expérience montrent que 1) la notion de résumé de référence qui sert aux évaluations classiques est discutable 2) les compressions semi-automatiques ont été fortement appréciées 3) les compressions totalement automatiques ont également obtenu de bons scores de satisfaction. À un taux de compression supérieur à 40 % tous genres confondus, CoLin fournit un support appréciable en tant qu'aide à la compression de textes, ne dépend d'aucun corpus d'apprentissage, et présente une interface convivial.

Mots clés : TALN ; résumé automatique ; résumé semi-automatique ; compression de phrases ; théorie du gouvernement et du liage ; arbre syntaxique ; grammaire de constituants ; outil interactif

---

Title: Automatic or semi-automatic text compression through removable constituent pruning: an interactive and corpus-free approach

Abstract: This research belongs to the Natural Language Processing field and more specifically focuses on text summarization. The originality of this thesis leads in tackling a type of summarization that has not been studied much, text compression using an unsupervised method. This work presents an interactive and incremental system for syntagmatic tree pruning, while preserving the syntactic coherence and the main informational contents. On the theoretical side, this work is based on the Government and Biding theory of Noam Chomsky and more precisely on the formal representation of the X-bar theory, to aims at a strong foundation for a computational model compatible with syntactic compression of sentences. This work led to an operational software, named CoLin, which proposes two modalities: an automated compression and an assistance to summarization in a semi-automated form, directed through a tight interaction with the user. This software has been evaluated thanks to a quite complex protocol using 25 volunteers. Experiment results show that 1) the notion of reference abstract which is the basic of classical evaluation is at least questionable, 2) semi-automated compression has been given a high value by users 3) fully automated compressions also get honourable satisfaction levels. With a compression ratio of over 40 % for all genres of text, CoLin offers an appreciable support as an assistance to text compression, without resorting on a learning corpus, and with a user-friendly interface.

Keywords: NLP; automatic summarization; semi-automatic summarization; sentence compression; government and binding theory; syntactic tree; constituent grammar; interactive tool

---

Discipline : Informatique

Laboratoire : Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM) ; UMR 5506 ; 161 rue Ada, 34392 Montpellier Cedex 5, France