Exploitation des tables du Lexique-Grammaire pour l'analyse syntaxique automatique

Benoît Sagot [1], Elsa Tolone [2]
INRIA Paris-Rocquencourt / Paris 7, Université Paris-Est

Résumé

Dans cet article, nous montrons comment nous avons converti les tables du Lexique-Grammaire en un format TAL, celui du lexique Le*fff*, permettant ainsi son intégration dans l'analyseur syntaxique FRMG. Nous présentons les fondements linguistiques de ce processus de conversion et le lexique obtenu. Nous comparons les résultats de l'analyseur syntaxique FRMG sur le corpus de référence de la campagne EASy selon qu'il utilise les entrées verbales du Le*fff* ou celles des tables des verbes du Lexique-Grammaire ainsi converties.

Abstract

In this paper, we describe how we converted the Lexique-Grammaire tables of French verbs into an NLP format, that of the Le*fff* lexicon, which allowed us to integrate it into the FRMG parser. We describe the linguistic basis of this conversion process, and the resulting lexicon. We compare the results of the FRMG parser on the EASy reference corpus depending on whether it relies on the verb entries of the Le*fff* or those of the converted Lexicon-Grammar verb tables.

Mots-clés : Lexiques syntaxiques, Lexique-Grammaire, analyse syntaxique, évaluation.

1. Introduction

Les tables du Lexique-Grammaire constituent aujourd'hui une des principales sources d'informations lexicales syntaxiques pour le français. Leur développement a été initié dès les années 1970 par Maurice Gross, au sein du LADL puis de l'IGM (Université Paris-Est) (Gross 1975; Boons *et al.* 1976; Guillet & Leclère 1992). Ces informations se présentent sous la forme de *tables*. Chaque table regroupe les éléments d'une catégorie donnée partageant certaines *propriétés définitoires*, qui relèvent généralement de la sous-catégorisation. Ces éléments forment une *classe*. Une table se présente sous forme de matrice : en lignes, les éléments lexicaux de la classe correspondante ; en colonnes, les propriétés qui ne sont pas forcément respectées par tous les éléments de la classe ; à la croisée d'une ligne et d'une colonne le signe $+$ ou $-$ selon que l'entrée lexicale décrite par la ligne accepte ou non la propriété décrite par la colonne. Il existe notamment 61 tables de verbes simples.

Les tables actuelles souffrent de diverses formes d'incohérence et d'incomplétude. En particulier, les propriétés définitoires ne sont pas représentées dans les tables [3]. Pour y remédier, des *tables des classes* sont en développement à l'IGM pour chaque catégorie, et notamment les verbes, qui associent à chaque classe l'ensemble de ses propriétés définitoires (Paumier 2003). Les résultats préliminaires de ce travail de fond nous ont permis de convertir les tables en un

[1] ALPAGE, INRIA Paris-Rocquencourt / Paris 7, benoit.sagot@inria.fr

[2] IGM, Université Paris-Est, elsa.tolone@univ-paris-est.fr

[3] Ceci a également motivé les travaux de (Gardent *et al.* 2005). Nous renvoyons à (Constant & Tolone 2008) pour une comparaison entre la version texte des tables, utilisée ici, et le travail présenté par (Gardent *et al.* 2005).

format qui permette leur utilisation dans un analyseur syntaxique à grande échelle, l'analyseur FRMG (Thomasset & Éric de La Clergerie 2005). C'est l'objet du présent article.

2. Le lexique verbal *lglex*

Une table des classes regroupe en colonnes l'ensemble de toutes les propriétés syntaxiques répertoriées pour la catégorie concernée, et liste en lignes l'ensemble des classes définies pour cette même catégorie. À l'intersection d'une ligne et d'une colonne, le symbole $+$ (resp. $-$) indique que la propriété correspondante est vérifiée (resp. non vérifiée) par tous les éléments de la classe (c'est-à-dire par toutes les entrées de la table correspondante). Le symbole *o* indique que la propriété est explicitement codée dans la table concernée, car elle est vérifiée par certaines de ses entrées mais pas toutes. Enfin, le symbole ? indique une cellule non encore renseignée.

Le développement de la table des classes des verbes et de celle des noms est bien avancé (Constant & Tolone 2008), seules les propriétés non encore étudiées pour certaines classes restent codées ?. Grâce à ce travail de mise en cohérence et d'explicitation des propriétés syntaxiques des verbes dans les tables du Lexique-Grammaire, il a été possible de construire une version structurée des tables du Lexique-Grammaire, disponible en un format textuel et en un format XML, et appelé lexique *lglex* (Constant & Tolone 2008) [4]. La construction de *lglex* repose sur l'outil *LGExtract*, qui prend en entrée les tables d'une catégorie donnée, la table des classes de cette catégorie et un fichier de configuration. Ce fichier définit comment chaque propriété (issue de la table des classes ou, dans le cas de propriétés qui y sont codées *o*, de la table correspondante) contribue à la construction de l'entrée *lglex*.

C'est à partir des entrées verbales de la version texte du lexique *lglex* que nous avons effectué une conversion vers le format du Le*fff*, le format Alexina.

3. Le Le*fff* et le format Alexina

Le Le*fff* (Lexique des formes fléchies du français) est un lexique syntaxique à large couverture pour le français (Sagot *et al.* 2006; Sagot & Danlos 2007) [5]. Il repose sur l'architecture Alexina d'acquisition et de modélisation de lexiques morphologiques et syntaxiques. Le lexique *intensionnel* associe à chaque entrée un cadre de sous-catégorisation canonique et liste les redistributions possibles à partir de ce cadre. Le processus de *compilation* du lexique intensionnel en lexique extensionnel construit différentes entrées pour chaque forme fléchie du lemme et chaque redistribution possible. Soit l'entrée intensionnelle simplifiée suivante :

clarifier$_1$ *Lemma;v;*<*arg*$_0$*:Suj:cln|scompl|sinf|sn,arg*$_1$*:Obj:(cla|scompl|sn)*>*;*
%ppp_employé_comme_adj,%actif,%se_moyen_impersonnel,
%passif_impersonnel,%passif

Elle décrit une entrée du lemme verbal *clarifier*, qui est transitive directe (deux arguments réa-

[4] Distribution partielle en ligne sous license LGPL-LR à l'adresse `http://infolingu.univ-mlv.fr`

[5] Distribution en ligne sous license LGPL-LR à l'adresse `http://gforge.inria.fr/projects/alexina/`

lisés canoniquement par les fonctions syntaxiques Suj et Obj décrites entre les chevrons), et qui admet les redistributions fonctionnelles *participe passé employé comme adjectif*, *actif* (la distribution par défaut), *se moyen impersonnel*, *passif impersonnel*, et *passif*.

Les fonctions syntaxiques sont définies dans le Le*fff* par des critères proches de ceux de DICOVALENCE (van den Eynde & Mertens 2006). L'inventaire des fonctions syntaxiques est le suivant : Suj (sujet), Obj (objet direct), Objà (objet indirect introduit canoniquement par la préposition *à*), Objde (objet indirect introduit canoniquement par la préposition *de*), Loc (locatif), Dloc (délocatif), Att (attribut), Obl ou Obl2 (autres arguments obliques). Les critères définitoires de ces fonctions sont décrits dans (Sagot & Danlos 2007).

Chaque fonction syntaxique peut être réalisée par différentes réalisations, qui sont de trois types : *pronoms clitiques*, *syntagme direct* (syntagme nominal (sn), adjectival (sa), infinitif (sinf), phrastique fini (scompl), interrogative indirecte (qcompl)) et *syntagme prépositionnel* (syntagme direct précédé d'une préposition, comme de-sn, à-sinf ou pour-sa ; à-scompl et de-scompl représentent les réalisations en *à/de ce que P*). Enfin, une fonction dont la réalisation est facultative voit sa liste de réalisations possibles mise entre parenthèses.

Des informations syntaxiques complémentaires (contrôle, mode des complétives, etc.) sont notées par des *macros* (@CtrlSujObj, @ComplSubj, etc.) dont l'interprétation formalisée dépend du contexte d'utilisation. Une modélisation de ces macros en LFG est fournie avec le Le*fff*.

4. Conversion du lexique verbal *lglex* en un lexique au format Alexina

Cette section décrit brièvement le processus de conversion au format Le*fff* du lexique verbal *lglex*. Ce processus est décrit plus en détail dans (Sagot & Tolone 2009).

Chaque entrée de *lglex* est associée à des constructions dont on peut distinguer plusieurs types :

1. la (ou les) construction(s) « de base », définitoires de la classe d'origine de l'entrée ;
2. les constructions « de base étendues », obtenues par adjonction d'arguments à la construction de base ; en pratique, ces constructions sont toutes des intermédiaires entre la construction de base et une construction dite « de base maximalement étendue » ou CBME ;
3. les constructions qui sont des variantes de la construction de base, obtenues par effacement d'un ou de plusieurs arguments ou par changement de type de réalisation (Qu P devenant V^{i}inf W, par exemple) ;
4. les constructions qui sont des redistributions ([passif de], N_1 est Vpp de ce Qu P) ;
5. les constructions dont il semble qu'elles auraient dû conduire à des entrées distinctes, dites « entrées secondaires », telles que les constructions neutres ou les transformations de type N_1 se V de ce Qu P (cf. *Luc se félicite d'avoir réussi à séduire Léa* vs. *Max félicite Luc qu'il ait réussi à séduire Léa*) ;

Nous avons développé une méthode permettant d'*aligner* deux constructions, c'est-à-dire de

construire des correspondances entre arguments, malgré leurs différences de surface [6] et leur possible effacement. Cette méthode nous permet d'identifier et d'aligner la CBME et ses variantes, rassemblées pour former une entrée unique du lexique final, dite entrée *canonique*. Parmi les autres constructions, celles qui correspondent à des redistributions standard ([passif par], [extrap]...) induisent l'ajout à l'entrée finale de la redistribution correspondante. Les autres induisent la création d'une entrée distincte, qu'elle relève du cas 5 ou qu'elle corresponde à une redistribution non encore répertoriée par le format Alexina.

Une fois répertoriées les entrées à produire, les cadres de sous-catégorisation sont construits. Pour cela, on construit d'abord le cadre correspondant à la construction maximale de chaque entrée (la CBME pour l'entrée canonique, ou l'unique construction des entrées secondaires). Des heuristiques permettent de définir la fonction syntaxique de chaque argument [7]. Leurs réalisations sont construites en deux temps. Tout d'abord, le type de syntagme (nominal, infinitif, phrastique,...) est déterminé. Ceci étant fait, les introducteurs possibles sont répertoriés à partir de l'ensemble des prépositions et autres introducteurs (p.ex. *et*). Dans le cas de l'entrée canonique, les différentes variantes de la CBME induisent des modifications du cadre de sous-catégorisation obtenue, en rajoutant des réalisations et en rendant certains arguments facultatifs.

D'autres types d'informations sont ensuite ajoutées pour former l'entrée finale, telles que la table d'origine et le numéro d'entrée dans cette table ou encore une indication de fréquence issue du DELA. Enfin, des macros syntaxiques concernant l'auxiliaire, le mode des complétives arguments, les clitiques figés (*se, en, ne*...) et les phénomènes de contrôle sont extraits et ajoutés à l'entrée finale.

Le lexique verbal obtenu contient 16 903 entrées pour 5 694 lemmes verbaux différents (2,96 entrées par lemme en moyenne). À titre de comparaison, le Le*fff* contient seulement 7 072 entrées verbales pour 6 818 lemmes verbaux distincts (1,04 entrées par lemme). Le lexique issu de *lglex*, quoique décrivant moins de lemmes verbaux, est donc beaucoup plus couvrant en termes de constructions syntaxiques et donc beaucoup plus ambigu. Au niveau extensionnel, le Le*fff* contient 361 268 entrées, alors que le lexique extrait de *lglex* en contient 763 555.

5. Intégration dans l'analyseur syntaxique FRMG

L'objectif premier de ce travail est de permettre aux données linguistiques codées dans les tables du Lexique-Grammaire de servir de base de données lexicales pour un analyseur syntaxique

[6] Par exemple, Qu P vs. N1, ou encore à N1 vs. Prép N1 si l'on sait par ailleurs que la Prép peut être *à*.

[7] Les fonctions syntaxiques sont obtenues de la façon suivante. Tout d'abord, le premier argument reçoit toujours la fonction Suj. Le premier argument post-verbal, s'il est direct, reçoit la fonction Obj, sauf pour les entrées de la table 32NM. Ensuite, un argument introduit par *à* (resp. *de*) reçoit la fonction syntaxique Objà (resp. Objde), sauf si un indice complémentaire vient contredire ce choix (par exemple, pour un argument N_1 introduit par *à*, la propriété à N_1 = Ppv = : le lui confèrera fonction syntaxique Obj, ex. : *Il apprend à conduire / Il l'apprend*). Les arguments introduits par *Loc* ont la fonction syntaxique Loc, sauf ceux de la forme Loc N_i source ou vérifiant Loc N_i = : de N_i source, qui ont la fonction syntaxique Dloc. Enfin, les autres arguments sont considérés comme des Att s'ils sont directs, et comme des Obl s'ils sont introduits par une préposition (Obl2 si un Obl existe déjà).

automatique du français. Parmi les analyseurs qui prennent en entrée un lexique au format Le*fff*, nous avons choisi l'analyseur FRMG (Thomasset & Éric de La Clergerie 2005). Il s'appuie sur une Grammaire d'Adjonction d'Arbres (TAG) compacte du français générée à partir d'une métagrammaire, et sur le Le*fff*. La compilation et l'exécution de l'analyseur se déroule dans le cadre du système DyALog (de La Clergerie 2005). Il utilise comme entrée le résultat de la chaîne de traitement présyntaxique SxPipe (Sagot & Boullier 2008).

L'intégration du lexique au format Le*fff* extrait de *lglex* dans l'analyseur FRMG est immédiate : le *lexeur* de l'analyseur fait appel à une base de données lexicales construite à partir du Le*fff*. Il suffit de remplacer les entrées verbales du Le*fff* par le lexique construit à partir de *lglex*, de conserver les autres entrées du Le*fff*, de construire la base de données lexicales correspondantes, et de spécifier à FRMG d'utiliser cette dernière. Le résultat est une variante de l'analyseur FRMG, que nous noterons FRMG_{lglex}, par opposition à la variante standard notée $\text{FRMG}_{\text{Le}fff}$.

6. Évaluation et discussion

Nous avons évalué $\text{FRMG}_{\text{Le}fff}$ et FRMG_{lglex} en analysant la partie annotée manuellement du corpus EASy (Paroubek *et al.* 2005), soit 4 306 phrases de styles variés (journalistique, médical, oral, questions, littéraire…). Nous avons utilisé les métriques définies et utilisées à l'occasion de la première campagne EASy d'évaluation des analyseurs syntaxiques, qui a eu lieu fin 2005 (Paroubek *et al.* 2006).

Avant de discuter des résultats de cette expérience, certaines précautions sont à prendre :

– le processus de conversion décrit ici et son implémentation encore préliminaire contiennent certainement des erreurs, et nous évaluons FRMG lorsqu'il utilise les entrées verbales converties à partir des tables, et non pas les entrées telles qu'elles sont dans les tables ;
– le Le*fff* a été développé en parallèle aux campagnes EASy, contrairement aux tables ; certains choix faits dans le guide d'annotation EASy ont pu influencer certains choix faits dans le développement du Le*fff*, alors que ce n'est évidemment pas le cas pour les tables ;
– pour cette expérience, *lglex* a été complété par diverses entrées verbales venant du Le*fff*, qui ne font pas partie du lexique *lglex* : entrées pour les auxiliaires et semi-auxiliaires, certains verbes à montée, les verbes impersonnels et les entrées pour les têtes syntaxiques des constructions à verbes support ; il se peut que d'autres entrées soient encore à rajouter.

Les résultats comparatifs sur les chunks et les « relations » EASy (dépendances entre mots pleins) entre ces deux analyseurs sont donnés à la table 1, avec le détail pour quelques sous-corpus illustratifs. Les résultats sont donc pour l'instant légèrement meilleurs pour $\text{FRMG}_{\text{Le}fff}$. Nous ne pensons pas que ce résultat remette en question la pertinence de l'utilisation des tables du Lexique-Grammaire en analyse syntaxique, notamment au vu des précautions ci-dessus. En particulier, nous restons convaincus que l'utilisation d'une ressource lexicale aussi riche que possible reste un moyen efficace pour améliorer la qualité d'un analyseur syntaxique, comme l'ont montré par exemple la mise en œuvre des travaux décrits dans (Sagot & de La Clergerie

	Chunks		Relations	
Sous-corpus	FRMG$_{\text{Le}fff}$	FRMG$_{lglex}$	FRMG$_{\text{Le}fff}$	FRMG$_{lglex}$
general_lemonde	**86.8%**	82.8%	**59.8%**	56.9%
general_senat	82.7%	**83.1**%	**56.7%**	54.9%
litteraire_2	**84.7%**	81.5%	**59.2%**	56.3%
medical_2	85.4%	**89.2%**	**62.4%**	58.6%
oral_delic_8	**74.1%**	73.6%	47.2%	**48.5%**
questions_amaryllis	90.5%	**90.6%**	**65.6%**	63.2%
total	***84.4%***	*82.3%*	***59.9%***	*56.6%*

TAB. *1. Résultats comparatifs EASy de* FRMG$_{\text{Le}fff}$ *et* FRMG$_{lglex}$.

2008). On peut toutefois constater que les temps d'analyse sont plus de deux fois plus élevés avec FRMG$_{lglex}$ qu'avec FRMG$_{\text{Le}fff}$ (temps médian par phrase de 0,62 s contre 0,26 s), ce qui provient certainement du nombre d'entrées par lemme qui est trois plus élevé dans *lglex* que dans le Le*fff*, comme vu plus haut. Du reste, ce temps d'analyse plus élevé conduit nécessairement à un plus grand nombre d'échecs d'analyse par dépassement du délai maximum autorisé, ce qui conduit à la construction d'analyses partielles, nécessairement de moins bonne qualité.

Sur certains sous-corpus, c'est toutefois FRMG$_{lglex}$ qui obtient les meilleures évaluations en chunks. Cependant, les résultats sur les relations sont moins bons avec FRMG$_{lglex}$, à l'exception de deux sous-corpus. L'analyse des résultats montre les faits suivants :

– FRMG$_{lglex}$ donne de meilleurs résultats que FRMG$_{\text{Le}fff}$ pour certaines relations, comme « modifieur d'adjectif » et « modifieur d'adverbe », mais également deux relations pour lesquels les résultats sont mauvais d'un côté comme de l'autre (« modifieur de préposition » et « apposition ») ;
– la relation « attribut » (du sujet ou de l'objet) est celle pour lequel la différence en rappel est la plus importante (34,0% contre 58,4%) ;
– le degré d'ambiguïté lexicale bien plus élevé dans FRMG$_{lglex}$ que dans FRMG$_{\text{Le}fff}$ conduit à un taux d'ambiguïté plus grand au niveau de l'analyseur, et donc d'autant plus de risque de se tromper au moment de la désambiguïsation. En effet, le désambiguïsateur utilisé par FRMG est à base de règles heuristiques pondérées [8] ;
– par exemple, le nombre élevé d'arguments verbaux listés dans les cadres de sous-catégorisation de *lglex* conduit à induire en erreur l'heuristique de désambiguïsation habituelle selon laquelle « on préfère les arguments aux modifieurs » : tout syntagme pouvant être analysé comme un argument verbal a tendance à l'être. Ainsi, dans une phrase comme *[...] on estime que cette décision [ferait] dérailler le processus de paix*, FRMG$_{lglex}$ fait de *de paix* un argument de *estimer* (*estimer qqch de qqn*), là où FRMG$_{\text{Le}fff}$ ne se trompe pas.

À court terme, on peut faire le constat suivant. De nombreuses phrases reçoivent une analyse complète par FRMG$_{lglex}$ mais pas par FRMG$_{\text{Le}fff}$, et inversement. Par exemple, sur le sous-corpus

[8] On peut penser qu'un modèle statistique de désambiguïsation pourrait mieux gérer l'ambiguïté provenant de *lglex*. Nous ne sommes pas en mesure de confirmer ou d'infirmer empiriquement cette hypothèse.

`general_lemonde`, 177 phrases sont entièrement reconnues par les deux analyseurs, 85 seulement par FRMG$_{\text{Le}fff}$, 76 seulement par FRMG$_{lglex}$, et 111 par aucun des deux. Or l'expérience montre que la qualité des résultats est très supérieure, de plus de 10 points en moyenne, sur les phrases analysées complètement par rapport à celles qui reçoivent plusieurs analyses partielles. On peut donc imaginer construire un système très simple qui analyserait une phrase avec chacun des deux analyseurs, et qui, si seulement l'un des deux propose une analyse complète, ne garde que cette analyse (ce qu'il faut faire dans les autres cas reste à étudier). Les résulats sont susceptibles d'être meilleurs que chacun des deux systèmes.

À plus long terme, il importe de bénéficier de cette complémentarité entre les deux ressources. Il sera intéressant d'étudier les différences entre les erreurs faites par chacun de ces deux analyseurs, y compris au moyen de techniques automatiques telles que celles décrites dans (Sagot & de La Clergerie 2008). Ceci pourrait permettre d'améliorer les différentes ressources, voire de détecter automatiquement des erreurs dans les lexiques. Dans le cas de *lglex*, nous nous attendons à ce que la plupart de ces erreurs proviennent du processus de conversion, mais certaines proviendront peut-être d'erreurs dans les tables du Lexique-Grammaire, et permettront donc d'améliorer ces dernières.

Bibliographie

Boons J.-P., Guillet A. et Leclère C. (1976), *La structure des phrases simples en français, Constructions intransitives*, Droz, Genève, Suisse.

Constant M. et Tolone E. (2008), « A generic tool to generate a lexicon for NLP from Lexicon-Grammar tables », in : *Actes du 27ème Colloque Lexique et Grammaire*, L'Aquila, Italie.

de La Clergerie É. (2005), « DyALog : a Tabular Logic Programming based environment for NLP », in : *Proceedings of 2nd International Workshop on Constraint Solving and Language Processing (CSLP'05)*, Barcelone, Espagne.

Gardent C., Guillaume B., Perrier G. et Falk I. (2005), « Maurice Gross' Grammar Lexicon and Natural Language Processing », in : *Proceedings of the 2nd Language and Technology Conference*, Poznań, Pologne.

Gross M. (1975), *Méthodes en syntaxe*, Hermann, Paris, France.

Guillet A. et Leclère C. (1992), *La structure des phrases simples en français : Les constructions transitives locatives*, Droz, Genève, Suisse.

Paroubek P., Pouillot L.-G., Robba I. et Vilnat A. (2005), « EASy : campagne d'évaluation des analyseurs syntaxiques », in : *Actes de l'atelier EASy de TALN 2005*, Dourdan, France.

Paroubek P., Robba I., Vilnat A. et Ayache C. (2006), « Data, Annotations and Measures in EASy, the Evaluation Campaign for Parsers of French », in : *Proceedings of the 5th international conference on Language Resources and Evaluation (LREC 2006)*, Gênes, Italie.

Paumier S. (2003), *De la reconnaissance de formes linguistiques à l'analyse syntaxique*, Université Paris-Est Marne-la-Vallée, [Thèse de Doctorat].

Sagot B. et Boullier P. (2008), « SXPipe 2 : architecture pour le traitement présyntaxique de corpus bruts », in : *Traitement Automatique des Langues (T.A.L.)*, n° 2, vol. 49, À paraître.

Sagot B., Clément L., de La Clergerie É. et Boullier P. (2006), « The Le*fff* 2 syntactic lexicon for French : architecture, acquisition, use », in : *Proceedings of the 5th Language Resource and Evaluation Conference*, Lisbonne, Portugal.

Sagot B. et Danlos L. (2007), « Améliorer un lexique syntaxique à l'aide des tables du lexique-grammaire – Constructions impersonnelles », in : *Cahiers du Cental*.

Sagot B. et de La Clergerie É. (2008), « Fouille d'erreurs sur les sorties d'analyseurs syntaxiques », in : *Traitement Automatique des Langues (T.A.L.)*, n° 1, vol. 49.

Sagot B. et Tolone E. (2009), « Intégrer les tables du Lexique-Grammaire à un analyseur syntaxique », in : *Actes de TALN'09 (session posters)*, Senlis, France.

Thomasset F. et Éric de La Clergerie (2005), « Comment obtenir plus des Méta-Grammaires », in : *Proceedings of TALN'05*, Dourdan, France.

van den Eynde K. et Mertens P. (2006), *Le dictionnaire de valence* DICOVALENCE *: manuel d'utilisation*, <http ://bach.arts.kuleuven.be/dicovalence/manuel_061117.pdf>.